DIGITAL ACCESS TO SCHOLARSHIP AT HARVARD
DASH.HARVARD.EDU

HARVARD LIBRARY
Office for Scholarly Communication

Genome Engineering Technologies to Change the Genetic Code

Citation

Lajoie, Marc Joseph. 2014. Genome Engineering Technologies to Change the Genetic Code. Doctoral dissertation, Harvard University.

Permanent link

http://nrs.harvard.edu/urn-3:HUL.InstRepos:11745697

Terms of Use

This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Other Posted Material, as set forth at http://nrs.harvard.edu/urn-3:HUL.InstRepos:dash.current.terms-of-use#LAA

Share Your Story

The Harvard community has made this article openly available.
Please share how this access benefits you. Submit a story .

Accessibility

Genome Engineering Technologies to Change the Genetic Code

A dissertation presented

by

Marc Joseph Lajoie

to

The Committee on Higher Degrees in Chemical Biology

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the subject of

Chemical Biology

Harvard University

Cambridge, Massachusetts

December 2013

© 2013 Marc J. Lajoie
All rights reserved.

Dissertation Advisor: George M. Church Marc J. Lajoie

Genome Engineering Technologies to Change the Genetic Code

ABSTRACT

New technologies are making it possible to engineer organisms with fundamentally new and useful properties. *In vivo* genome engineering technologies capable of manipulating genomes from the nucleotide to the megabase scale were developed and applied to reassign the genetic code of *Escherichia coli*. Such genomically recoded organisms show promise for thwarting horizontal gene transfer with natural organisms, resisting viral infection, and expanding the chemical properties of proteins.

Chapter 1 discusses the boundaries of possible genetic codes and the barriers that must be overcome to produce them.

Chapters 2 and 3 describe mechanistically-driven improvements to λ Red recombineering, which is a cornerstone of our approach to genome engineering. Chapter 2 is adapted from J. A. Mosberg, et al. *Genetics* (2010) **186**, 791. Chapter 3 is adapted from portions of J. A. Mosberg, et al. *PLoS One* (2012) **7**, e44638 and M. J. Lajoie, et al. *Nucleic Acids Res.* (2012) **40**, e170.

Chapter 4 describes how to make hundreds of changes in a genome in order to reassign the UAG codon to new function. Multiplex automated genome engineering (MAGE) was used to site-specifically replace all known UAG stop codons with synonymous UAA codons in parallel across 32 *E. coli* strains. Conjugative assembly genome engineering (CAGE) was used to hierarchically merge these codon changes. This chapter is adapted from F. J. Isaacs, et al. *Science* (2011) **333**, 348.

Chapter 5 describes the complete reassignment of the UAG codon in *E. coli* and provides preliminary evidence that genetic codes can be changed to resist viruses and to expand the chemical diversity of proteins. This chapter is adapted from M. J. Lajoie, et al. *Science* (2013) **342**, 357.

Chapter 6 probes the limits of genetic recoding in *E. coli* by radically recoding 42 essential genes. Across 80 *E. coli* strains, all instances of 13 rare codons were removed from these genes and all remaining codons were shuffled as possible. The results suggest that *in vivo* genome engineering and genetic diversity will be essential for radically changing genetic codes. This chapter is adapted from M. J. Lajoie, et al. *Science* (2013) **342**, 361.

TABLE OF CONTENTS

ABSTRACT..... iii

TABLE OF CONTENTS..... v

LIST OF FIGURES..... vii

LIST OF TABLES..... ix

DEDICATION..... x

ACKNOWLEDGEMENTS..... xi

CHAPTER 1
Introduction: Engineering Genetic Codes..... **1**

CHAPTER 2
Lambda Red Recombination in Escherichia coli Occurs Through a Fully Single-Stranded Intermediate..... **28**

CHAPTER 3
Manipulating Replisome Dynamics and DNA Exonucleases to Enhance Lambda Red-Mediated Multiplex Genome Engineering..... **51**

CHAPTER 4
Genome-wide Codon Replacement Using Synthetic Oligonucleotides and Engineered Conjugation..... **87**

CHAPTER 5
Genomically Recoded Organisms Impart New Biological Functions..... **118**

CHAPTER 6
Probing the Limits of Genetic Recoding in Essential Genes..... **155**

CHAPTER 7
Conclusions and Future Projects..... **172**

APPENDIX A
Supplemental information for Chapter 2..... **177**

APPENDIX B
Supplemental information for Chapter 3..... **185**

APPENDIX C
Supplemental information for Chapter 4..... **200**

APPENDIX D
Supplemental information for Chapter 5**..... 224**

APPENDIX E
Supplemental information for Chapter 6**..... 302**

LIST OF FIGURES

Figure 1-1. Properties of genomically recoded organism (GROs) with a reassigned UAG codon..... **2**

Figure 1-2. Minimal and maximal genetic codes using triplet codons composed of four nucleotide types (U, C, A, G)..... **11**

Figure 2-1. Previously proposed Lambda Red-mediated dsDNA recombination mechanisms..... **32**

Figure 2-2. Lambda Red mediated dsDNA recombination proceeds via a ssDNA intermediate..... **33**

Figure 2-3. Strand bias in Lambda Red ssDNA insertion recombination..... **36**

Figure 2-4. Strand-specific mismatch alleles were used to identify the strand of origin for each recombined mutation..... **37**

Figure 2-5. Testing the effect of strand protection on recombination frequency..... **40**

Figure 3-1. AR optimization via CoS-MAGE..... **55**

Figure 3-2. Effect of dnaG attenuation on replication fork dynamics..... **56**

Figure 3-3. Effect of Nuclease Genotype on CoS-MAGE Performance..... **59**

Figure 3-4. DnaG variants improve CoS-MAGE Performance..... **61**

Figure 3-5. Placing all targeted alleles within one Okazaki Fragment does not cause a bimodal distribution for recombination frequency..... **65**

Figure 3-6. Testing DnaG variants with a 20-plex CoS-MAGE oligo set..... **66**

Figure 3-7. Averaged CoS-MAGE performance by strain..... **73**

Figure 4-1. Strategy for reassigning all 314 UAG codons to UAA in *E. coli*..... **91**

Figure 4-2. Frequency map of oligo-mediated UAG::UAA codon replacements and genetic marker integrations across the *E. coli* genome at each replacement position..... **93**

Figure 4-3. Clonal rate and distribution of genome modifications after 18 cycles of MAGE... **94**

Figure 4-4. Hierarchical CAGE methodology for controlled genome transfer..... **97**

Figure 5-1. Engineering a GRO with a reassigned UAG codon..... **121**

Figure 5-2. Effects of UAG reassignment at natural UAG codons..... **123**

Figure 5-3. NSAA incorporation in GROs..... **124**

Figure 5-4. Bacteriophage T7 infection is attenuated in GROs lacking RF1..... **127**

Figure 6-1. Codon reassignment across 42 essential genes..... **157**

Figure 6-2. Recoded strain doubling times..... **159**

Figure 6-3. Schematics of all changes introduced in recoded essential genes..... **161**

Figure 7-1. Proposed future genetic code reassignment..... **173**

LIST OF TABLES

Table 2-1. Tracking co-segregation in mismatched dsDNA recombination..... **38**

Table 3-1. Summary of mean number of alleles converted per clone for each MAGE oligo set..... **62**

Table 3-2. CoS-MAGE Allele Replacement performance of modified strains (presented as fold change from EcNR2)..... **63**

Table 5-1. Recoded strains and their genotypes..... **126**

DEDICATION

This thesis is dedicated to my family,

for all of your support and love

ACKNOWLEDGEMENTS

I can't imagine doing my PhD anywhere but in the Church lab. Professor George Church has put together an amazing team of people, who have become great collaborators and friends. Personally, George gave me the opportunity to plan my own projects, write my own papers and grants, and start my own collaborations. He helped facilitate unique opportunities like my trip to Lindau and my month in the Söll lab, and he always made time for me when I needed advice or a pertinent story about his past experiences. George truly cares most about science, and he is extremely aggressively selfless ("selfless" is not a typo) in this endeavor. George is one of the people who I most respect in this world—scientifically and personally.

When I first joined the Church lab, people used to joke with me that Farren Isaacs was my boss. He taught me how to do genome engineering research, mentored me on rE. coli, and certainly gave me more assignments than George did, but he always treated me like a friend and a collaborator. Even though Farren has left to start his own lab, I'm grateful for his continued mentorship and friendship.

I also have to recognize Dieter Söll and Lanny Ling for mentoring me on the biochemistry of the genetic code, which has given me valuable perspective in thinking about how to change it. In addition, Steve Elledge, Brian Seed, and David Rudner served on my Dissertation Advisory Committee, and provided helpful feedback and advice throughout my time in the Church lab.

Josh Mosberg and Chris Gregg are my baymates, and we have collaborated on just about everything that we have touched since joining the Church lab. These guys are my go-to guys when I have an idea, a problem, or just need a break.

Sri Kosuri is an amazing scientist and collaborator who always has an answer, whether about science, career, or life.

Although Dan Mandell joined the team toward the end of my time in the Church lab, it has been a real pleasure working with him—bartering genome engineering expertise for protein design expertise and bartering peanut butter protein bars for fudge brownie protein bars.

Sara Vassallo was a great friend and PCR helper. Most of all, though, I loved her infectious happiness that made the Church lab a brighter place.

Tara Gianoulis was like a big sister to me. She was a role model for how to squeeze the most out of life and how to be a good scientist. I will miss her for the rest of my life.

There's no way that I could have accomplished all of this work without my collaborators and co-authors: Dan Goodman, Gleb Kuznetsov, Michael Napolitano, Matthieu Landon, Harris Wang, Peter Carr, Mike Mee, Di Zhang, Joanne Ho, Arthur Sun, Jaron Mercer, Gabriel Washington, Lakshmi Govindarajan, Xavier Rios, Lexi Rovner, Jesse Rinehart, Hans-Rudolf Aerni, Nadin Rohland, Andy Tolonen, Marc Güell, Julie Norville, Mark Moosburner, Wei Leong Chew, Jun Teramoto, and Barry Wanner.

In addition to the considerable list of collaborators, I have to thank a few more people who spent a considerable amount of time teaching and/or helping me with only my gratitude as payment: Prashant Mali, Uri Laserson, Raj Chari, Poyi Huang, Mike Sismour, Nikolai Eroshenko, Kevin Esvelt, and Jacob Carlson. Finally, I want to thank the rest of the Church lab, Genetics Department, and Wyss Institute. In soccer, I always improve the most when I'm playing with people who are better than I am. This has been equally relevant in my PhD training. In this respect, the Church lab has been an amazing place. I've managed to connect with just about everyone in the lab at some point, whether it was for brainstorming, experimental advice,

or brewing recombinant beer. I've learned a lot from all of you, and I want you to know how much I appreciate the privilege of interacting with you over the past 5 years.

Outside of the lab, I still can't escape the lab—some of my best friends spend 10 hours next to me at the bench before we move the party elsewhere. When I do manage to escape the lab culture, Nate Wicksman, DJ Travers, Dave Lindenbaum, and Andrew Jean-Louis make sure that I hang out, and my soccer teams have done their part in keeping me balanced and healthy both mentally and physically.

Finally, I am incredibly lucky to have such an amazing, loving, supportive family. My parents were my first teachers and have always been my biggest supporters. My favorite escape from Boston is to relax at home with my parents and three brothers Matt, Ben, and Nate. They have been an extremely important part of who I am, and I can't thank them enough for all of their love and support. Additionally, I've been extremely fortunate to marry in to an incredibly loving family. I knew that I was really a part of the family when Sarah lent me her car and Mom and Dad Rocio hosted me while I was at Yale and Kim was working in Boston.

Most importantly, I thank my wife, Kimberly, who has slogged through all of the sacrifices of my PhD with a smile on her face. She makes me eat, sleep, shave, and save time for a little bit of fun (and my mom appreciates this, too). She is always the best part of my day.

CHAPTER 1

Introduction: Engineering Genetic Codes

Acknowledgements:

G. Church, D. Söll, J. Ling, M. Sismour, and D. Mandell provided helpful discussion.

Introduction

The canonical genetic code has been good to us. For decades, biotechnology has relied on it to permit the transgenic production of drugs (*1*), materials (*2*), and food (*3*). However, the canonical genetic code also supports viruses (*e.g.*, HIV, influenza) and undesired horizontal gene transfer [*e.g.*, antibiotic resistance (*4*) and dissemination of recombinant DNA (*5*-*7*)]. Furthermore, its mere 20 amino acids stifle the potential for evolving new and useful protein functions.

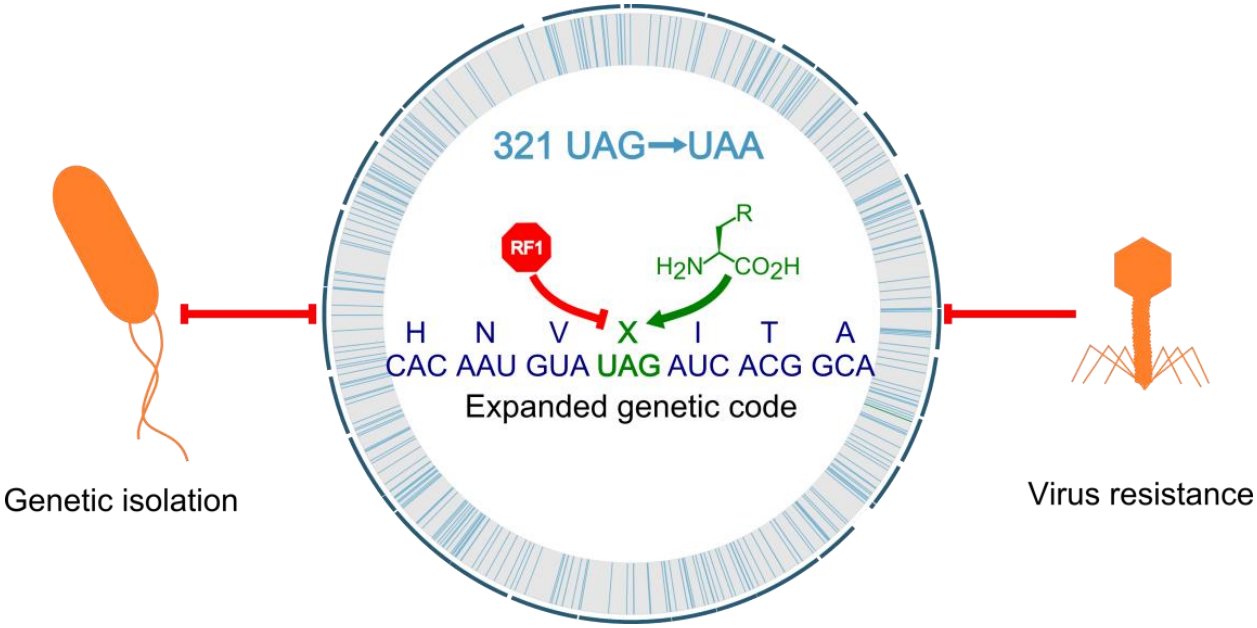


Figure 1-1. Properties of genomically recoded organism (GROs) with a reassigned UAG codon. The GRO provides a dedicated UAG codon for plug-and-play translation of nonstandard amino acids (NSAAs). This enables efficient expression of GFP variants containing several UAG codons, provides increased resistance to bacteriophages, and establishes a basis for the genetic isolation of GROs (*8*).

Genomically recoded organisms (GROs) possessing alternate genetic codes (*8*) have the potential to solve these problems (Figure 1-1). By interpreting genetic information differently, GROs would mistranslate foreign genes based on the canonical genetic code. This would prevent viruses from hijacking their translation machinery and thwart the transfer of functional genetic information with natural organisms. In addition, GROs could be engineered to incorporate more than 70 structurally diverse nonstandard amino acids (NSAAs) that have been developed to

enhance enzyme activity (*9, 10*), to improve the performance of protein drugs (*11*), and to function as molecular probes (*12*). Redesigning essential proteins to depend on NSAAs for proper translation and function would provide a robust strategy for restricting undesired survival outside of controlled conditions. Thus, GROs have the potential to be safe and powerful chassis for biofermentation, bioremediation, and agriculture: Virus resistance could save hundreds of millions of dollars from lost batches (*13, 14*), genetic isolation can reduce the risk of escape into the environment, and NSAAs can improve enzyme functions.

There are considerable biological and technological challenges that must be addressed in order to engineer such organisms with new genetic codes, and this review will focus on overcoming these barriers. While principles relevant to the evolution of the genetic code will be applied to this analysis, a more comprehensive discussion of how the genetic code originated can be found in previous work (*15-19*).

Central dogma of molecular biology and protein translation

A number of recent reviews explain the molecular details of the central dogma of molecular biology (*20, 21*) and protein translation (*22*). Nevertheless, a brief overview of these topics provides important exposition for engineering genetic codes.

Central dogma

Crick (*23*) stated that nucleic acid sequence information can be transferred to nucleic acids and proteins, whereas proteins cannot transfer sequence information (*23*). A simplistic view of the central dogma is as follows: Modern, free-living organisms possess double stranded DNA genomes that are copied completely during replication. Additionally, small portions of

DNA composed of one or more genes can provide the information to produce RNA during transcription. RNA can either function as a ribozyme (*e.g.*, ribosomal RNA, rRNA) or it can provide the information to produce proteins during translation (*i.e.*, messenger RNA, mRNA). Proteins are highly efficient catalysts that perform most of the cellular chemistry.

Protein translation

Aminoacyl-tRNA synthetases (aaRSs) are proteins that charge the correct amino acid onto the correct transfer RNAs (tRNAs). Each aaRS recognizes specific identity elements (*24*) on its target tRNA and has a binding pocket that determines which amino acid it charges. Each tRNA has an anticodon composed of three ribonucleotides that determine which codons it decodes. An elongation factor (EF-Tu) helps shuttle aminoacyl-tRNAs into the decoding center of the ribosome where the anticodon base pairs with a complementary three-ribonucleotide codon on the mRNA, which provides the sequence information for the protein being translated. Correct base pairing results in ribosome-catalyzed transfer of the amino acid onto the nascent peptide chain. Translation termination is performed with specialized proteins called release factors instead of tRNAs.

Protein translation is extremely complex and energy-intensive, revealing the importance of its accuracy for modern life (*25*). To accomplish this, an aaRS must charge the correct amino acid onto the correct tRNA, and the correct aminoacyl-tRNA must pair with the correct codon in the ribosomal A site. Ultimately, 3 base pairs (between 6 and 9 H-bonds) introduce the correct amino acid 10^3-10^4 times for every error (*25*).

Evolutionary barriers to reassignment

With a few notable exceptions, the genetic code is conserved across all three domains of life (*26*). Understanding this remarkable stability is essential to overcoming it.

Modern organisms have large genomes and require accurate translation

Evolution increases biological complexity (*27*), leading to the large genomes of today's free-living organisms [the smallest known genome is 580,070 base pairs, with 470 predicted coding regions (*28*)]. With a few exceptions (*29*-*31*), these organism use all 64 codons to encode their proteins and to accommodate overlapping non-coding motifs such as protein binding sites, promoters, splicing signals, and RNA secondary structure (*32*). Any change in codon function must be tolerated at all instances genome-wide. Furthermore, these larger and more complex genomes experience more structural constraints (*e.g.*, overlapping features in polycystronic operons), a larger mutational load, and a higher demand for translation fidelity (*33*-*35*). Given that modern translation systems appear to have improved accuracy compared to primordial systems (*26*), modern proteomes may have traded increased activity for a reduced tolerance for mistranslation (*26*, *33*).

The error minimization theory for the origin of the genetic code proposes that similar amino acids are grouped with similar codons, promoting mutational robustness (*i.e.*, single nucleotide mutations are likely to incorporate the same amino acid or a similar one) (*34*), increasing translation accuracy (*i.e.*, codon/anticodon mispairing most likely introduces the same amino acid or a similar one) (*33*), and accommodating non-coding information (*i.e.*, redundant code maintains protein sequence and allows flexibility for overlapping non-coding motifs) (*32*).

The canonical code does this remarkably well (*16*, *36*), utilizing all 64 codons for translation throughout the proteome, providing a disincentive for genetic code expansion (*37*).

Anticodons are assigned to specific amino acids

The stereochemical theory for the origin of the genetic code proposes that amino acids make direct chemical interactions with their cognate tRNAs (*38*). Regardless of whether this was the case, most amino acids are effectively associated with specific anticodons in the canonical genetic code through their aaRSs. With the exceptions of LeuRS, SerRS, and AlaRS, all other aaRSs in *E. coli* recognize tRNA anticodons as tRNA identity elements (*24*). This means that mutations in anticodons do not necessarily reassign the cognate codon function—in fact, the mutated tRNA may lose its recognition by the original aaRS and gain recognition from the aaRS corresponding to the new anticodon (*39*). Therefore, even if a codon is available for reassignment, the new tRNA must maintain its desired function and escape recognition from competing aaRSs.

Collaboration is advantageous

Horizontal gene transfer (*40*) and sexual reproduction (*41*) allow organisms to share beneficial traits and to remove deleterious traits on a population level. These processes can only happen if interacting organisms speak the same genetic language, providing a strong evolutionary incentive to maintain a common genetic code (*42*).

Lessons from naturally noncanonical genetic code

Despite substantial evolutionary pressures to conserve the genetic code, naturally noncanonical genetic codes [extensively reviewed (*18, 26, 43-45*)] exist, and it is likely that they all derive from the same canonical code (*43*). Studying natural codon reassignment can provide insight into how the genetic code evolves and how to synthetically change it. Interestingly, many of the same changes appear to have independently evolved several times, suggesting that certain codons have a predisposition for reassignment (*26*). Stop codons may be favored because they are only used once at the end of genes, so their reassignment is expected to cause minimal damage to the proteome (*26*). More generally, small tweaks to anticodon modifications that change codon assignment without affecting aaRS recognition account for most of the genetic code variation (*e.g.*, loss of lysidine from $tRNA^{Met}_{CAU}$ allows it to decode both AUG and AUA as Met and a 7-methylguanosine modification on $tRNA^{Ser}_{GCU}$ allows it to decode all four AGN codons) (*26*).

Now we know the easiest targets for codon reassignment, but how do natural organisms overcome the evolutionary barriers in order to change their function? The codon capture theory proposes that codons are eliminated from entire genomes prior to reassignment (*46*), and the ambiguous intermediate theory proposes that codons may initially introduce multiple amino acids until a selective pressure causes fixation of the new function (*47*). While significant evidence exists for each mechanism, it seems likely that elements of both mechanisms are relevant—a small number of codons may remain prior to codon capture (*39*), and ambiguous decoding must be tolerated at all instances genome-wide. Regardless of the mechanism, the change must provide a substantial selective advantage to offset reduced mutational robustness, translation fidelity, and horizontal gene transfer.

Implementation of codon capture

Small genomes (*48*), especially those with extreme biases in GC content (*17*), provide opportunities for codon capture to occur. Mitochondria appear to have a strong selective pressure for small, AT rich genomes, which can lead to the spontaneous loss of codons (*26*). Subsequently, anticodon modifications increase the promiscuity of codon recognition, allowing a single tRNA to decode multiple codons (*26*). Examples of free-living organisms with noncanonical genetic codes include *Mycoplasma* species (*28*, *30*) and SR1 uncultured oral bacteria (*31*), which have their UGA stop codons reassigned to Trp and Gly, respectively. Both bacteria have small genomes and appear to strongly select for low G + C content (*31*). G + C bias is hypothesized to drive codon reassignment (Trp UGG→UGA and Gly GGA→UGA conversions help reduce G + C content), but these variations in the genetic code may also help reduce susceptibility to viruses (*31*, *49*).

Implementation of ambiguous intermediate

Larger genomes are less likely to spontaneously lose all instances of a given codon. Therefore, ambiguous decoding for a given codon must be tolerated at all instances genome-wide. Indeed *E. coli* tolerates natural suppressors of its stop codons (*47, 50*) and *C. ablicans* (*51*) decodes CUG codons as both Leu (canonical assignment) and Ser. Indeed, related fungi exhibit complete reassignment of CUG to encode Ser, suggesting that *C. albicans* may be a glimpse at an ambiguous intermediate (*51*). Therefore, spontaneous ambiguous intermediates can occur and evolve at the mercy of genetic drift.

What about when the 20 canonical amino acids are not adequate? While post-translational modifications can help to tune amino acid properties, two additional naturally-occurring amino acids have been identified, which are incorporated during translation (*26*). Organisms from all domains of life use selenocysteine (Sec) in essential redox enzymes, and *Methanosarcinaceae* use pyrrolysine (Pyl) in methanogenesis from methylamines (*26*). However, these organisms already use all 64 codons for translation, requiring them to reuse at least one codon for these specialized functions. In order to accomplish this, these organisms use orthogonal translation machinery that are dependent on unique recognition sequences in the mRNA to efficiently introduce Sec or Pyl at specific positions in target proteins (*26*). Therefore, Sec and Pyl are exciting examples of how natural selection added new chemical functionalities to the genetic code to expand protein function.

Minimal and maximal genetic codes

It is now clear that the genetic code continues to evolve, but what are its limits? It may be possible to add a new base pair (*52, 53*) or to engineer a quadruplet genetic code (*54-56*), which could give $6^3 = 216$ or $4^4 = 256$ possible codons, respectively. For this analysis, however, let's consider the minimal and maximal variants of the current genetic code, which possesses triplet codons composed of four possible nucleotides (Figure 1-2). Although these hypothetical genetic codes may be far from optimal and difficult to implement, it is instructive to consider the fundamental biochemical boundaries for the genetic code. In *E. coli*, 43 unique anticodons and release factors unambiguously decode all 64 codons (Figure 1-2A) (*57*). Codon recognition is controlled by base pairing between the codon (mRNA) and anticodon (tRNA), and post-transcriptional chemical modifications tune which base pairs are recognized (*57*). Therefore, it is

possible to significantly alter the genetic code by altering anticodons. Additionally, orthogonal aaRS/tRNA pairs can be introduced to expand the amino acid repertoire (*58*).

Minimal genetic code

A minimal genetic code requires one tRNA for each amino acid [including formylmethionine for translation initiation (*59*)] and one release factor for translational termination (Figure 1-2B). As demonstrated by mitochondrial genetic codes, an unmodified uracil in the anticodon wobble position can recognize all four codons that are identical at the first two positions and differ at the third position (*43*). In this way, 10 tRNAs are adequate to unambiguously assign 40 codons to decode 9 amino acids (in this example, Arg is decoded by two separate family groups, and one tRNA could potentially be deleted, leaving four blank codons). Anticodons with guanosine, queuosine, or glutamylqueuosine in the wobble position use 6 additional tRNAs that unambiguously recognize 12 codons of the form NNY (N = any of the four bases; Y = C or U) to translate 6 more amino acids (*19*). Additionally, mnm^5U modifications on 4 tRNAs unambiguously recognize 8 codons of the form NNR (R = G or A) to translate 4 more amino acids (*19*). This leaves all three stop codons, which can be recognized by a single E167K release factor 2 variant (*60*), in addition to Trp and translation initiation, which use their natural tRNAs. Finally, it may be possible to achieve adequate protein function using a code composed of fewer than 20 amino acids (*61-64*). Preliminary studies propose that Ile (*65*) and/or Trp (*66*) could be replaced by similar amino acids. Therefore, the minimal genetic code requires 23 tRNAs and a release factor in order to decode all 64 codons, or 20 tRNAs if Ile and Trp are removed and blank codons are tolerated (Figure 1-2B).

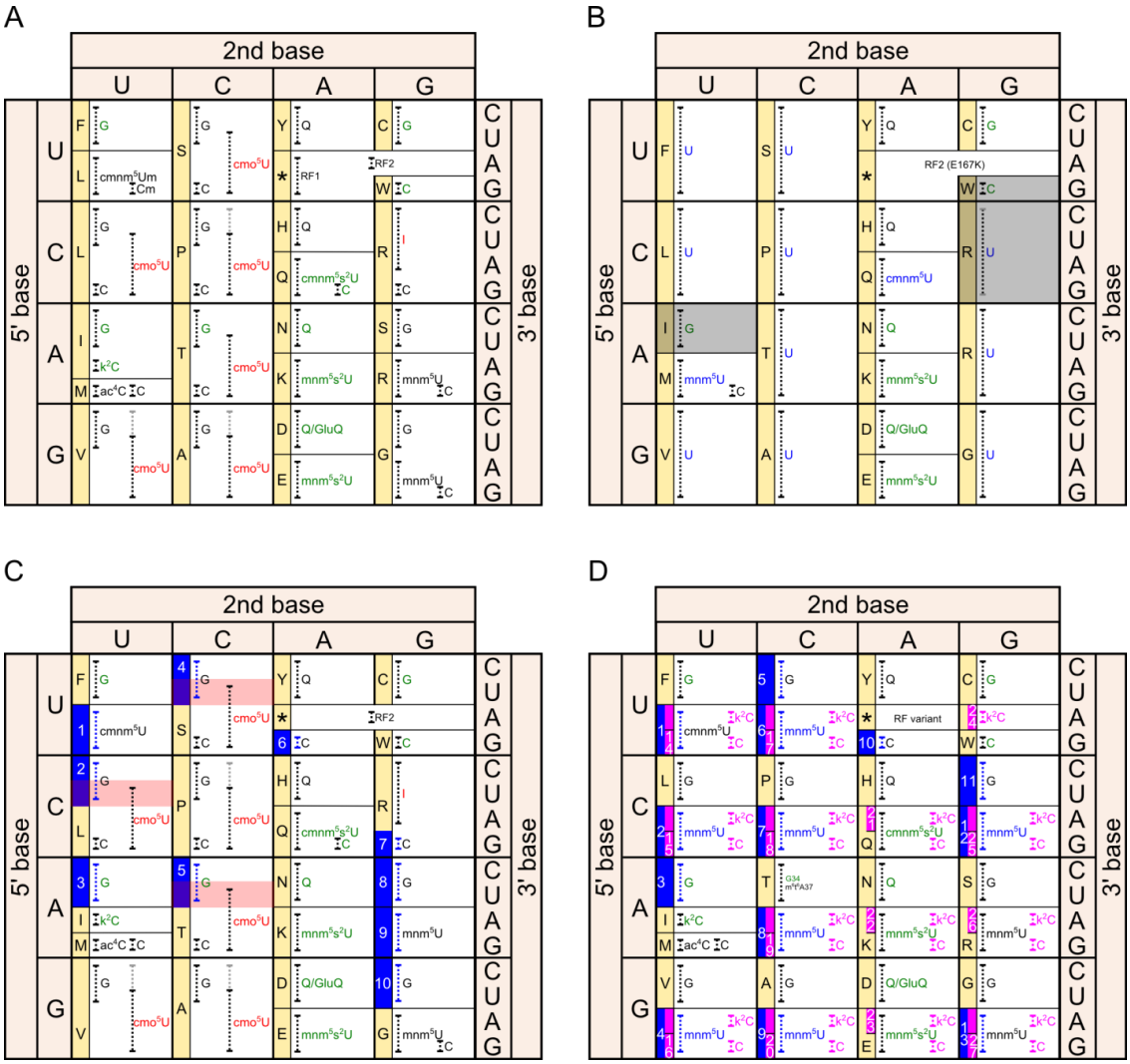


Figure 1-2. Minimal and maximal genetic codes using triplet codons composed of four nucleotide types (U, C, A, G). The proposed genetic codes are one possible permutation representing several possible ways to reassign redundant codons (*e.g.*, which of the six Ser codons should maintain Ser function after the others are reassigned). Dashed brackets represent anticodon – codon recognition ranges: black is codon recognition agreeing with wobble rules (*115, 116*); gray is empirical data (*57*); blue and magenta are new tRNAs assigned to new amino acids. Labels correspond to the wobble nucleotide at tRNA position 34 (cmo^5U = uridine 5-oxyacetic acid, mnm^5U = 5-methylaminomethyluridine, $cmnm^5U$ = 5-carboxymethylaminomethyluridine, $cmnm^5Um$ = 5-carboxymethylaminomethyl-2′-*O*-methyluridine, mnm^5s^2U = 5-methylaminomethyl-2-thiouridine, $cmnm^5s^2U$ = 5-carboxymethylaminomethyl-2-thiouridine, I = inosine, k^2C = lysidine, Q = queuosine, GluQ = glutamylqueuosine) (*117*). Green letters indicate natural tRNA identity determinants that may be difficult to change. Red letters indicate natural anticodon modifications that increase anticodon promiscuity. Blue and magenta letters represent proposed changes in the tRNA wobble position that would alter codon recognition. Amino acid assignments are indicated in the yellow sidebars. M refers to Met and fMet (translation initiation). Anticodons available for new amino acids are blue and magenta boxes with white numbers. Selenocysteine is not shown. **(A)** The *E. coli* genetic code is presented based on Björk *et al.* (*57*) and tRNA identity determinants are from Giegé *et al.* (*24*). All 64 codons are used to encode 20 amino acids. **(B)** A minimal genetic code utilizing all 64 codons would require initiation at AUG, one release factor [RF2 E167K mutants can terminate all 3 stop codons (*60*)], and one tRNA for each of the 20 amino acids. Unmodified uracils in the wobble positions would allow tRNAs to recognize all codons in a family group, allowing redundant tRNAs to be deleted. Gray shaded boxes represent additional anticodons that could be potentially deleted [$tRNA^{Arg}_{UCG}$ would encode the same amino acid as $tRNA^{Arg}_{UCU}$ and it may be possible to remove Ile (*65*) and/or Trp (*66*) from the genetic code]. Conveniently, the wobble nucleotide is rarely a tRNA identity determinant (*24*). The

Figure 1-2 (Continued). two relevant exceptions, $tRNA^{Phe}_{GAA}$ (G34) and $tRNA^{Glu}_{UUG}$ ($cmnm^5s^2U34$), are weak identity determinants (*24*), so the proposed changes may be tolerated by their respective aaRSs. **(C)** The genetic code can be expanded to provide 7 unambiguous and 3 ambiguous anticodons by simply deleting tRNAs and introducing orthogonal aaRS/tRNA pairs encoding new amino acids. This analysis assumes that the original aaRS/tRNA identity determinants/antideterminants can be overcome by a metagenomic search for an orthogonal aaRS/tRNA pair and subsequent directed evolution to optimize their orthogonality. Red shaded boxes represent the three codons that would gain ambiguous translation function upon introduction of an orthogonal aaRS/tRNA pair. The UAG codon can be liberated by deleting release factor 1 (*8*). **(D)** The maximal genetic code would have unique amino acid assignments for all NNA and NNG codons [NNA: engineer *tilS* (*70*) to lysidinylate additional tRNAs so that they only base pair with A. NNG: tRNAs with cytosine in the wobble position only base pair with G]. NNY codons could not be split into unambiguous NNU and NNC codons using known anticodon modifications, but such modifications have not been ruled out. Additionally, it may be possible to engineer a release factor to terminate translation only at UAA codons, thereby liberating both UAG and UGA codons. The proposed changes would liberate 27 unambiguous anticodons (47 total amino acids; changes indicated in magenta). This strategy may require directed evolution to overcome the tRNA identity determinants for Glu, Gln, and Lys (*24*). Another potential complicating factor is that G + C anticodon content may affect cognate and near-cognate decoding efficiencies, just as the G + C rich anticodons for Val, Ala, and Pro break the wobble rules (*57*). More conservatively, replacing the cmo^5U [inactivate *cmoB* (*67*)] and inosine wobble nucleotides with mnm^5U nucleotides [engineer *mnmE* and *mnmG* to recognize additional tRNAs (*68, 69*)] could liberate 13 unambiguous anticodons (33 total amino acids; changes indicated in blue).

Maximal genetic code

Expanding the genetic code requires unassigned codons that can be appropriated for new functions and orthogonal aaRS/tRNA pairs that can carry out those functions without having cross-talk with endogenous ones. For now, we will focus on capturing codons; orthogonal translation machinery will be discussed below. Simply by leveraging the degeneracy of genetic code, 10 anticodons can be deleted in order to provide 7 unambiguous and 3 ambiguous anticodons for reassignment, while maintaining translation initiation, termination, and incorporation of all 20 amino acids (Figure 1-2C). Inconveniently, six family groups (CUN, GUN, UCN, CCN, ACN, and GCN) are composed of anticodons with overlapping codon specificity, making it difficult to unambiguously reassign their function. The overlapping tRNA specificities are caused by cmo^5U wobble bases, which are able to base pair with A, G, and U (and sometimes C) (*67*). In contrast, the GGN codon family group uses a mnm^5U wobble base to decode only GGA and GGG, allowing unambiguous reassignment of GGY to a new function. This strategy could be extended to other tRNAs by inactivating *cmoB* (modifies U34 of tRNAs with cmo^5U) (*67*), and engineering *mnmE* and *mnmG* to modify U34 of additional tRNAs with

mnm^5 (naturally modifies $tRNA_{Gln}$, $tRNA_{Lys}$, $tRNA_{Glu}$, and $tRNA_{Arg}$) (*68*, *69*). These modifications could capture 13 unambiguous anticodons (up to 33 total amino acids) (Figure 1-2C, blue features).

More aggressively, the NNR codons could be split into unique singlet codons by exploiting anticodons modified with lysidine (specifically base pairs with A) and cytosine (specifically base pairs with G) to decode NNA and NNG codons, respectively (Figure 1-2D, magenta features). In order to accomplish this, *tilS* would need to be engineered to lysidinylate more anticodons in addition to its natural target, $tRNA^{Ile}$ (*70*). Although wobble codons do not tend to coincide with tRNA identity determinants, this could affect the aminoacylation of the tRNAs for Glu, Gln, and Lys because mnm^5-modified wobble bases are minor tRNA identity determinants (*24*). The NNY codons cannot be split into singlet codons based on known anticodon modifications, but such modifications have not been ruled out. Finally, a release factor could potentially be engineered to terminate translation at UAA codons, but not UAG or UGA codons. These proposed changes could capture 27 unambiguous anticodons (up to 47 total amino acids).

How do we change the genetic code?

Knight *et al.* describe the problem well: "a novel code must be both chemically plausible and mutationally accessible from its immediate ancestor" (*43*). We have now seen that vastly simplified and expanded codes are possible from biochemical principles, but the implementation of such codes has many evolutionary caveats that must be addressed. Removing the redundancy of the canonical genetic code would sacrifice mutational robustness, translational fidelity, and sequence flexibility used to accommodate non-coding information (*e.g.*, polycystronic operons

would require refactoring in order to move regulatory motifs outside of genes). Such modified genetic codes would require robust selective pressures to maintain their new functions despite these countervailing evolutionary forces. Additionally, orthogonal translation machinery must be provided in order to reassign codons to new amino acids. Finally, the evolutionary incentive for horizontal gene transfer must be obviated by providing optimal cultivation conditions for the recoded organism. By taking these principles into account, remarkable progress has already been made at expanding the genetic code to incorporate more than 70 NSAAs (*12*).

In vitro translation

In vitro translation systems offer the ultimate flexibility to implement translation (*71, 72*). *In vitro* systems provide unique opportunities to use biologically incompatible chemistry to prepare component parts. By performing aminoacyl-tRNA charging separately, CA ligation (*73*) and flexizyme (*74*) can abstract away the need for orthogonal aaRS/tRNA pairs to charge tRNAs. Additionally, *in vitro* systems do not need to support essential cellular functions, so suboptimal translation components are better tolerated, and the evolutionary dependence on the canonical genetic code is vastly reduced. The extreme genetic codes proposed in Figure 1-2 could be readily tested in an *in vitro* system. To date, *in vitro* translation has been the best way to produce synthetic nonribosomal peptide mimetics (*75*) and non-peptide polymers such as polyesters (*76*). However, *in vitro* systems can be expensive, complicated, and difficult to scale for industrial applications.

In vivo suppression of natural codon function

In vivo systems are well-suited for inexpensive, simple, and scalable translation using NSAAs. In the absence of efficient genome engineering technologies to mimic the codon capture mechanism for codon reassignment, early *in vivo* approaches took advantage of an ambiguous intermediate (*77*). Sense codons have been transiently diverted to incorporate diverse NSAAs by metabolic labeling (*78*). In the cases of some NSAAs, this ambiguous intermediate has become well-tolerated. While bacteriophages can rapidly evolve tolerance for ambiguous incorporation 6-fluorotryptophan in place of Trp (*79*), results have been mixed for autonomous organisms, which express a larger pool of proteins that may be disrupted by NSAA incorporation. For example, while *B. subtilis* (*80*) has been evolved to prefer 4-fluorotryptophan (4fp) over tryptophan (Trp), similar experiments have been less successful in *E. coli (81)*.

The ambiguous intermediate strategy becomes even more difficult when reassigning codons to incorporate structurally distinct nonstandard amino acids (NSAAs). In such cases, NSAA exclusion may provide a simpler survival mechanism than evolving tolerance for NSAA incorporation (*82*). Recognizing these constraints, the Schultz, Chin, Wang, Liu, Söll, Neumann, and Ellington labs have made several advances in engineering the genetic code at the translation level, enabling the incorporation of more than 70 NSAAs into proteins [extensively reviewed (*12, 37, 83*)]. The general strategy is to introduce an orthogonal aaRS/tRNA pair that is evolved to specifically incorporate a NSAA without cross-charging endogenous aaRS/tRNA pairs. Orthogonal aaRS/tRNA pairs must have tRNA identity determinants that differ from those in the target organism. Unfortunately, this is difficult to accomplish for most codons in *E. coli* because most of its tRNAs have identity determinants in their anticodons. For instance, a heterologous $tRNA^{Pyl}_{CCG}$ was mischarged with Arg by the *E. coli* ArgRS, presumably due to recognition of the anticodon (*39*). In contrast, suppressors of the UAG stop codon have been more successful in

the absence of an aaRS that recognizes the CUA anticodon. For this reason, the weak cross-charging of *M. jannaschii* TyrRS/$tRNA_{CUA}$ with *E. coli* aaRS/tRNA pairs was readily overcome using directed evolution (*58*). Additional work will be required to identify an effective aaRS/tRNA pair for each of the remaining codons (*37*), and preliminary work with the orthogonal selenocysteine translation machinery provides preliminary success at reassigning 58 of the 64 codons (*84*). The *E. coli* tRNA identity determinants have been extensively characterized (*24*), and their antideterminants have been predicted (*85*), providing a starting point for the directed evolution of additional orthogonal aaRS/tRNA pairs.

Even when orthogonal aaRS/tRNA pairs are available, they must compete with highly optimized endogenous translation machinery (*37*). Therefore, suppression of rare codons like UAG (*58*) and over-expression of the orthogonal aaRS/tRNA pairs (*86*) are crucial for efficient NSAA incorporation. To take this to an extreme, attenuating UAG termination substantially increased the efficiency of NSAA incorporation (*87, 88*). In addition, quadruplet codons can provide additional channels for NSAA incorporation (*54, 55, 89*), and orthogonal ribosomes [reviewed in (*56*)] have been engineered to more efficiently decode UAG (*90*) and AGGA (*55*) codons. Other innovations include the addition of *p*-aminophenylalanine biosynthetic machinery to create an autonomous bacterium that uses a genetic code composed of 21 amino acids (*91*), and unnatural nucleoside base pairs to create a 65^{th} codon (*53*).

In vivo reassignment of natural codon function

How do you implement new amino acids in stable and heritable ways? Ambiguous codon suppression methods are most effective for single-batch overexpression of NSAA-containing proteins. Furthermore, while preliminary work has successfully incorporated two NSAAs into

the same protein, competition with natural codon functions reduces yields (*54, 55, 92*). Therefore, it would be beneficial to completely reassign codon function by removing endogenous translation factors before replacing them with new ones; however, this would result in deleterious mistranslation at all natural instances in the proteome. This mistranslation can be avoided by replacing all instances of the target codon with a synonymous one prior to reassignment. At a minimum, all instances of a target codon in essential genes should be changed to a synonymous codon (recoded) in order to preserve essential cellular functions (*93*). This strategy is a shortcut to codon capture, but it risks deleterious, proteome-scale misfolding in response to particularly disruptive NSAAs such as phosphoserine (*8*). Therefore, a more general and scalable solution to codon reassignment would be to recode all instances of a codon genome-wide (*8*). After removing all instances of a given codon, its translation would no longer be necessary for normal proteome function, allowing the removal of its natural translation factors and the introduction of a new translation function. This strategy has been used to completely reassign UAG from a stop codon to a sense codon (*8*). Furthermore, preliminary evidence suggests that this strategy could be extended to capture 12 additional codons for reassignment (*94*).

Genome engineering methods for changing the genetic code

The past decade has seen many impressive achievements in genome engineering [reviewed in (*95*)], although few attempts have been made introduce synthetic sequences that cannot be found in nature. The *de novo* synthesis and transplantation of an intact *Mycoplasma mycoides* JCVI-syn1.0 genome demonstrated that a small, natural prokaryotic genome can be built from simple chemical components (*96*). Such an approach could allow the synthesis of any

user-defined genome sequence, but genome design remains the major barrier. Whole genomes are a risky engineering unit because a single, cryptic design flaw could cause them to fail. For example, considerable effort was required to identify and correct a single base pair deletion in essential gene *dnaA*, initially preventing the transplantation of *M. mycoides* JCVI-syn1.0 (*96*). Given the high stakes for design flaws, *de novo* genome synthesis is most effectively used for projects based on exhaustive empirical tests (*28, 97, 98*) and complete computational models (*99*).

Engineering the genetic code requires extensive genome manipulation that can affect fitness in unpredictable ways (*94*). With this in mind, our lab has developed multiplex automated genome engineering (MAGE) (*100*) and conjugative assembly genome engineering (CAGE) (*101*) for rapidly prototyping and manufacturing genotypes *in vivo*. MAGE uses synthetic ssDNA oligonucleotides and the phage λ Red β recombinase (*102*) to simultaneously introduce defined mutations at multiple locations throughout a replicating bacterial genome (*100*). CAGE uses bacterial conjugation to precisely transfer up to several million base pairs of contiguous DNA (*101*), allowing the assembly of large genomes from small segments that are easier to produce and test using MAGE. Together, MAGE and CAGE exploit evolution to combinatorially explore a broad pool of synthetically defined genotypes *in vivo*, allowing natural selection to remove deleterious design flaws from the population.

MAGE and CAGE were used to remove all 321 known instances of the UAG codon from *E. coli* MG1655 at a fraction of the predicted cost for genome synthesis (*8*). However, DNA synthesis can still be effective for radically changed genome sequences (*94*). When thousands of changes are required genome synthesis becomes a more attractive strategy, and chip-based DNA synthesis is dramatically reducing its costs (*103-105*). In fact, small, synthetic genome segments

synergize well with MAGE-based troubleshooting of potential design flaws (*94*). Indeed this approach has also been successful for the synthetic yeast 2.0 project (*106*), and similar approaches could be extended to diverse organisms using an ever-growing arsenal of powerful genome engineering methods (*107*).

Outlook and conclusions

While more than 70 NSAAs have already vastly expanded protein function (*12*), radically different genetic codes will be required to achieve virus resistance, genetic isolation, and stable expansion of the genetic code. This will require orthogonal translation machinery (*84*, *108*-*111*) that are engineered to reassign sense codons based on a solid mechanistic understanding of biochemical principles (*37*). It will also require the design and construction of viable genomes with thousands of potentially deleterious changes in order to capture codons for reassignment (*8*, *94*). Advances in understanding codon usage (*112*) and operon structure (*113*, *114*) will help establish better guidelines for genome design, but diversity will remain a crucial aspect in prototyping genomes with new and useful biological functions.

References

1. D. V. Goeddel *et al.*, Expression in Escherichia coli of chemically synthesized genes for human insulin. *PNAS* **76**, 106 (1979).

2. C. E. Nakamura, G. M. Whited, Metabolic engineering for the microbial production of 1,3-propanediol. *Curr. Opin. Biotechnol.* **14**, 454 (Oct, 2003).

3. S. R. Padgette *et al.*, Development, identification, and characterization of a glyphosate-tolerant soybean line. *Crop Sci.* **35**, 1451 (Sep-Oct, 1995).

4. H. C. Neu, The crisis in antibiotic resistance. *Science* **257**, 1064 (Aug, 1992).

5. B.-R. Lu, A. A. Snow, Gene flow from genetically modified rice and its environmental consequences. *BioScience* **55**, 669 (2005/08/01, 2005).

6. A. Harris, D. Beasley, Bayer agrees to pay $750 million to end lawsuits over gene-modified rice, <http://www.bloomberg.com/news/2011-07-01/bayer-to-pay-750-million-to-end-lawsuits-over-genetically-modified-rice.html>. *Bloomberg*, (July 2, 2011, 2011).

7. H.-B. Xia, W. Wang, H. Xia, W. Zhao, B.-R. Lu, Conspecific Crop-Weed Introgression Influences Evolution of Weedy Rice (*Oryza sativa* f. *spontanea*) across a Geographical Range. *PLoS One* **6**, e16189 (2011).

8. M. J. Lajoie *et al.*, Genomically recoded organisms expand biological functions. *Science* **342**, 357 (Oct 18, 2013).

9. J. C. Jackson, S. P. Duffy, K. R. Hess, R. A. Mehl, Improving nature's enzyme active site with genetically encoded unnatural amino acids. *Journal of the American Chemical Society* **128**, 11124 (Aug, 2006).

10. I. N. Ugwumba *et al.*, Improving a natural enzyme activity through incorporation of unnatural amino acids. *Journal of the American Chemical Society* **133**, 326 (Jan, 2011).

11. H. Cho *et al.*, Optimized clinical performance of growth hormone with an expanded genetic code. *Proceedings of the National Academy of Sciences*, (May 16, 2011, 2011).

12. C. C. Liu, P. G. Schultz, Adding new chemistries to the genetic code. *An. Rev. Biochem.* **79**, 413 (2010).

13. J. M. Sturino, T. R. Klaenhammer, Engineered bacteriophage-defence systems in bioprocessing. *Nat. Rev. Microbiol.* **4**, 395 (2006).

14. V. Bethencourt, Virus stalls Genzyme plant. *Nat Biotech* **27**, 681 (2009).

15. R. D. Knight, S. J. Freeland, L. F. Landweber, Selection, history and chemistry: the three faces of the genetic code. *Trends in Biochemical Sciences* **24**, 241 (1999).

16. S. J. Freeland, R. D. Knight, L. F. Landweber, L. D. Hurst, Early fixation of an optimal genetic code. *Mol. Biol. Evol.* **17**, 511 (April 1, 2000, 2000).

17. E. V. Koonin, A. S. Novozhilov, Origin and evolution of the genetic code: the universal enigma. *IUBMB Life* **61**, 99 (Feb, 2009).

18. G. R. Moura, J. A. Paredes, M. A. S. Santos, Development of the genetic code: Insights from a fungal codon reassignment. *FEBS Lett.* **584**, 334 (2010).

19. P. T. S. van der Gulik, W. D. Hoff, Unassigned Codons, Nonsense Suppression, and Anticodon Modifications in the Evolution of the Genetic Code. *J. Mol. Evol.* **73**, 59 (Oct, 2011).

20. C. Bustamante, W. Cheng, Y. X. Meija, Revisiting the Central Dogma One Molecule at a Time. *Cell* **144**, 480 (Feb, 2011).

21. G. W. Li, X. S. Xie, Central dogma at the single-molecule level in living cells. *Nature* **475**, 308 (Jul, 2011).

22. V. Ramakrishnan, Ribosome structure and the mechanism of translation. *Cell* **108**, 557 (Feb, 2002).

23. F. Crick, Central dogma of molecular biology. *Nature* **227**, 561 (1970).

24. R. Giegé, M. Sissler, C. Florentz, Universal rules and idiosyncratic features in tRNA identity. *Nucleic Acids Res.* **26**, 5017 (November 1, 1998, 1998).

25. H. Jakubowski, E. Goldman, Editing of errors in selection of amino acids for protein synthesis. *Microbiological Reviews* **56**, 412 (Sep, 1992).

26. A. Ambrogelly, S. Palioura, D. Soll, Natural expansion of the genetic code. *Nat Chem Biol* **3**, 29 (2007).

27. G. M. Edelman, J. A. Gally, Degeneracy and complexity in biological systems. *Proc. Natl. Acad. Sci. U. S. A.* **98**, 13763 (Nov, 2001).

28. C. M. Fraser *et al.*, The minimal gene complement of Mycoplasma genitalium. *Science* **270**, 397 (Oct, 1995).

29. T. Oba, Y. Andachi, A. Muto, S. Osawa, CGG - An unassigned or nonsense codon in Mycoplasma capricolum. *Proc. Natl. Acad. Sci. U. S. A.* **88**, 921 (Feb, 1991).

30. Y. Inagaki, Y. Bessho, S. Osawa, Lack of peptide-release activity responding to codon UGA in Mycoplasma capricolum. *Nucleic Acids Res.* **21**, 1335 (Mar, 1993).

31. J. H. Campbell *et al.*, UGA is an additional glycine codon in uncultured SR1 bacteria from the human microbiota. *Proc. Natl. Acad. Sci. U. S. A.* **110**, 5540 (Apr, 2013).

32. S. Itzkovitz, U. Alon, The genetic code is nearly optimal for allowing additional information within protein-coding sequences. *Genome Research* **17**, 405 (Apr, 2007).

33. C. R. Woese, On evolution of genetic code. *Proc. Natl. Acad. Sci. U. S. A.* **54**, 1546 (1965).

34. C. J. Epstein, Role of amino acid code and of selection for conformation in evolution of proteins. *Nature* **210**, 25 (1966).

35. F. H. C. Crick, Origin of the genetic code. *Journal of Molecular Biology* **38**, 367 (1968).

36. H. Buhrman *et al.*, A realistic model under which the genetic code is optimal. *J. Mol. Evol.* **77**, 170 (Oct, 2013).

37. P. O'Donoghue, J. Ling, Y.-S. Wang, D. Soll, Upgrading protein synthesis for synthetic biology. *Nature Chemical Biology* **9**, 594 (Oct, 2013).

38. M. Yarus, Amino acids as RNA ligands: A direct-RNA-template theory for the code's origin. *J. Mol. Evol.* **47**, 109 (Jul, 1998).

39. R. Krishnakumar *et al.*, Transfer RNA misidentification scrambles sense codon recoding. *Chembiochem* **14**, 1967 (Oct, 2013).

40. H. Ochman, J. G. Lawrence, E. A. Groisman, Lateral gene transfer and the nature of bacterial innovation. *Nature* **405**, 299 (May, 2000).

41. S. P. Otto, T. Lenormand, Resolving the paradox of sex and recombination. *Nat. Rev. Genet.* **3**, 252 (Apr, 2002).

42. K. Vetsigian, C. Woese, N. Goldenfeld, Collective evolution and the genetic code. *PNAS* **103**, 10696 (2006).

43. R. D. Knight, S. J. Freeland, L. F. Landweber, Rewiring the keyboard evolvability of the genetic code. *Nat. Rev. Genet.* **2**, 49 (Jan, 2001).

44. M. A. S. Santos, G. Moura, S. E. Massey, M. F. Tuite, Driving change: the evolution of alternative genetic codes. *Trends in genetics : TIG* **20**, 95 (2004).

45. K. Watanabe, S.-I. Yokobori, tRNA modification and genetic code variations in animal mitochondria. *Journal of nucleic acids* **2011**, 623095 (2011 (Epub 2011 Oct, 2011).

46. S. Osawa, T. H. Jukes, K. Watanabe, A. Muto, Recent evidence for evolution of the genetic code. *Microbiol. Mol. Biol. Rev.* **56**, 229 (March 1, 1992, 1992).

47. D. W. Schultz, M. Yarus, Transfer RNA mutation and the malleability of the genetic code. *Journal of Molecular Biology* **235**, 1377 (1994).

48. S. E. Massey, J. R. Garey, A comparative genomics analysis of codon reassignments reveals a link with mitochondrial proteome size and a mechanism of genetic code change via suppressor tRNAs. *J. Mol. Evol.* **64**, 399 (Apr, 2007).

49. D. C. Krakauer, V. A. A. Jansen, Red queen dynamics of protein translation. *J. Theor. Biol.* **218**, 97 (2002).

50. G. Eggertsson, D. Söll, Transfer ribonucleic acid-mediated suppression of termination codons in Escherichia coli. *Microbiological Reviews* **52**, 354 (September 1, 1988, 1988).

51. A. R. Bezerra *et al.*, Reversion of a fungal genetic code alteration links proteome instability with genomic and phenotypic diversification. *Proc. Natl. Acad. Sci. U. S. A.* **110**, 11079 (Jul, 2013).

52. J. Piccirilli, T. Krauch, S. Moroney, S. Benner, Enzymatic incorporation of a new base pair into DNA and RNA extends the genetic alphabet. *Nature* **343**, 33 (JAN 4 1990, 1990).

53. J. D. Bain, C. Switzer, R. Chamberlin, S. A. Benner, Ribosome-mediated incorporation of a nonstandard amino acid into a peptide through expansion of the genetic code. *Nature* **356**, 537 (APR 9 1992, 1992).

54. J. C. Anderson *et al.*, An expanded genetic code with a functional quadruplet codon. *Proc. Natl. Acad. Sci. U. S. A.* **101**, 7566 (May 18, 2004, 2004).

55. H. Neumann, K. Wang, L. Davis, M. Garcia-Alai, J. W. Chin, Encoding multiple unnatural amino acids via evolution of a quadruplet-decoding ribosome. *Nature* **464**, 441 (2010).

56. K. H. Wang, W. H. Schmied, J. W. Chin, Reprogramming the genetic code: from triplet to quadruplet codes. *Angew. Chem.-Int. Edit.* **51**, 2288 (2012).

57. G. R. Björk and T. G. Hagervall, posting date. Chapter 4.6.2. Transfer RNA modification. In Escherichia coli and Salmonella. *Cellular and Molecular Biology*. Edited by Curtiss III R., Böck A., Ingrahan J.L., Kaper J.B., Maloy S., Neidhardt F.C., Riley M.M., Squires C.L. and Wanner B.L. ASM Press., Washington DC. (2005).

58. L. Wang, A. Brock, B. Herberich, P. G. Schultz, Expanding the genetic code of Escherichia coli. *Science* **292**, 498 (Apr, 2001).

59. F. Sherman, J. W. Stewart, S. Tsunasawa, Methionine or not methionine at the beginning of a protein. *Bioessays* **3**, 27 (1985).

60. K. Ito, M. Uno, Y. Nakamura, Single amino acid substitution in prokaryote polypeptide release factor 2 permits it to terminate translation at all three stop codons. *Proceedings of the National Academy of Sciences* **95**, 8165 (July 7, 1998, 1998).

61. S. Akanuma, T. Kigawa, S. Yokoyama, Combinatorial mutagenesis to restrict amino acid usage in an enzyme to a reduced set. *Proc. Natl. Acad. Sci. U. S. A.* **99**, 13549 (Oct, 2002).

62. K. U. Walter, K. Vamvaca, D. Hilvert, An active enzyme constructed from a 9-amino acid alphabet. *J. Biol. Chem.* **280**, 37742 (Nov, 2005).

63. A. Kawahara-Kobayashi *et al.*, Simplification of the genetic code: restricted diversity of genetically encoded amino acids. *Nucleic Acids Res.* **40**, 10576 (Nov, 2012).

64. M. F. Lu *et al.*, Reconstructing a flavodoxin oxidoreductase with early amino acids. *Int. J. Mol. Sci.* **14**, 12843 (Jun, 2013).

65. V. Pezo *et al.*, Artificially ambiguous genetic code confers growth yield advantage. *Proc. Natl. Acad. Sci. U. S. A.* **101**, 8593 (June 8, 2004, 2004).

66. V. Pezo *et al.*, A metabolic prototype for eliminating tryptophan from the genetic code. *Sci Rep* **3**, (Feb, 2013).

67. S. J. Nasvall, P. Chen, G. R. Bjork, The modified wobble nucleoside uridine-5-oxyacetic acid in tRNA(cmo5UGG)(Pro) promotes reading of all four proline codons in vivo. *RNA-Publ. RNA Soc.* **10**, 1662 (Oct, 2004).

68. R. Shi *et al.*, Structure-function analysis of Escherichia coli MnmG (GidA), a highly conserved tRNA-modifying enzyme. *J. Bacteriol.* **191**, 7614 (December 15, 2009, 2009).

69. D. Pearson, T. Carell, Assay of both activities of the bifunctional tRNA-modifying enzyme MnmC reveals a kinetic basis for selective full modification of cmnm5s2U to mnm5s2U. *Nucleic Acids Res.* **39**, 4818 (June 1, 2011, 2011).

70. K. Nakanishi *et al.*, Structural basis for translational fidelity ensured by transfer RNA lysidine synthetase. *Nature* **461**, 1144 (Oct, 2009).

71. Y. Shimizu *et al.*, Cell-free translation reconstituted with purified components. *Nature Biotechnology* **19**, 751 (Aug, 2001).

72. A. C. Forster *et al.*, Programming peptidomimetic syntheses by translating genetic codes designed de novo. *Proc. Natl. Acad. Sci. U. S. A.* **100**, 6353 (May, 2003).

73. C. J. Noren, S. J. Anthonycahill, M. C. Griffith, P. G. Schultz, A general method for site-specific incorporation of unnatural amino acids into proteins. *Science* **244**, 182 (Apr, 1989).

74. H. Murakami, A. Ohta, H. Ashigai, H. Suga, A highly flexible tRNA acylation method for non-natural potypeptide synthesis. *Nat. Methods* **3**, 357 (May, 2006).

75. K. Josephson, M. C. T. Hartman, J. W. Szostak, Ribosomal synthesis of unnatural peptides. *Journal of the American Chemical Society* **127**, 11727 (Aug, 2005).

76. Fahnesto.S, A. Rich, Ribosome-catalyzed polyester formation. *Science* **173**, 340 (1971).

77. J. Bacher, R. Hughes, J. Wong, A. Ellington, Evolving new genetic codes. *Trends in Ecology & Evolution* **19**, 69 (2004).

78. J. T. Ngo, D. A. Tirrell, Noncanonical amino acids in the interrogation of cellular protein synthesis. *Accounts of chemical research* **44**, 677 (2011).

79. J. Bacher, J. Bull, A. Ellington, Evolution of phage with chemically ambiguous proteomes. *BMC Evolutionary Biology* **3**, (2003).

80. J. T. Wong, Membership mutation of the genetic code: loss of fitness by tryptophan. *Proceedings of the National Academy of Sciences* **80**, 6303 (October 1, 1983, 1983).

81. J. Bacher, A. Ellington, Selection and characterization of Escherichia coli variants capable of growth on an otherwise toxic tryptophan analogue. *J. Bacteriol.* **183**, 5414 (2001).

82. H. Hennecke, A. Bock, Altered alpha subunits in phenylalanyl-transfer-RNA synthetases from para-fluorophenylalanine-resistant strains of Escherichia coli. *European Journal of Biochemistry* **55**, 431 (1975).

83. H. Neumann, Rewiring translation - Genetic code expansion and its applications. *FEBS Lett.* **586**, 2057 (Jul, 2012).

84. M. J. Brocker, J. M. L. Ho, G. M. Church, D. Soll, P. O'Donoghue, Recoding the Genetic Code with Selenocysteine. *Angewandte Chemie International Edition* **32**, in press (2013).

85. D. H. Ardell, Computational analysis of tRNA identity. *FEBS Lett.* **584**, 325 (2010).

86. T. S. Young, I. Ahmad, J. A. Yin, P. G. Schultz, An enhanced system for unnatural amino acid mutagenesis in E. coli. *Journal of Molecular Biology* **395**, 361 (2009).

87. D. B. F. Johnson *et al.*, RF1 knockout allows ribosomal incorporation of unnatural amino acids at multiple sites. *Nat Chem Biol* **7**, 779 (2011).

88. I. L. Wu *et al.*, Multiple site-selective insertions of noncanonical amino acids into sequence-repetitive polypeptides. *Chembiochem* **14**, 968 (May, 2013).

89. T. J. Magliery, J. C. Anderson, P. G. Schultz, Expanding the genetic code: selection of efficient suppressors of four-base codons and identification of "shifty" four-base codons with a library approach in Escherichia coli. *Journal of Molecular Biology* **307**, 755 (2001).

90. K. Wang, H. Neumann, S. Y. Peak-Chew, J. W. Chin, Evolved orthogonal ribosomes enhance the efficiency of synthetic genetic code expansion. *Nature Biotechnology* **25**, 770 (Jul, 2007).

91. R. A. Mehl *et al.*, Generation of a bacterium with a 21 amino acid genetic code. *Journal of the American Chemical Society* **125**, 935 (2003/01/01, 2003).

92. W. Wan *et al.*, A facile system for genetic incorporation of two different noncanonical amino acids into one protein in Escherichia coli. *Angewandte Chemie International Edition* **49**, 3211 (2010).

93. T. Mukai *et al.*, Codon reassignment in the Escherichia coli genetic code. *Nucleic Acids Res.* **38**, 8188 (2010).

94. M. J. Lajoie *et al.*, Probing the limits of genetic recoding in essential genes. *Science* **342**, 361 (Oct 18, 2013).

95. P. A. Carr, G. M. Church, Genome engineering. *Nat Biotech* **27**, 1151 (2009).

96. D. G. Gibson *et al.*, Creation of a bacterial cell controlled by a chemically synthesized genome. *Science* **329**, 52 (Jul, 2010).

97. C. A. Hutchison *et al.*, Global transposon mutagenesis and a minimal mycoplasma genome. *Science* **286**, 2165 (Dec, 1999).

98. J. I. Glass *et al.*, Essential genes of a minimal bacterium. *Proc. Natl. Acad. Sci. U. S. A.* **103**, 425 (Jan, 2006).

99. J. R. Karr *et al.*, A whole-cell computational model predicts phenotype from genotype. *Cell* **150**, 389 (2012).

100. H. H. Wang *et al.*, Programming cells by multiplex genome engineering and accelerated evolution. *Nature* **460**, 894 (Aug, 2009).

101. F. J. Isaacs *et al.*, Precise manipulation of chromosomes in vivo enables genome-wide codon replacement. *Science* **333**, 348 (Jul, 2011).

102. H. M. Ellis, D. G. Yu, T. DiTizio, D. L. Court, High efficiency mutagenesis, repair, and engineering of chromosomal DNA using single-stranded oligonucleotides. *Proc. Natl. Acad. Sci. U. S. A.* **98**, 6742 (Jun, 2001).

103. J. D. Tian *et al.*, Accurate multiplex gene synthesis from programmable DNA microchips. *Nature* **432**, 1050 (Dec 23, 2004).

104. S. Kosuri *et al.*, Scalable gene synthesis by selective amplification of DNA pools from high-fidelity microchips. *Nat Biotech* **28**, 1295 (2010).

105. J. Y. Quan *et al.*, Parallel on-chip gene synthesis and application to optimization of protein expression. *Nature Biotechnology* **29**, 449 (May, 2011).

106. J. Dymond *et al.*, Synthetic chromosome arms function in yeast and generate phenotypic diversity by design. *Nature* **477**, 471 (2011).

107. K. M. Esvelt, H. H. Wang, Genome-scale engineering for systems and synthetic biology. *Mol. Syst. Biol.* **9**, (Jan, 2013).

108. R. A. Hughes, A. D. Ellington, Rational design of an orthogonal tryptophanyl nonsense suppressor tRNA. *Nucleic Acids Res.* **38**, 6813 (October 1, 2010, 2010).

109. H. Neumann, A. L. Slusarczyk, J. W. Chin, De novo generation of mutually orthogonal aminoacyl-tRNA synthetase/tRNA pairs. *Journal of the American Chemical Society* **132**, 2142 (2010/02/24, 2010).

110. A. Chatterjee, H. Xiao, P. G. Schultz, Evolution of multiple, mutually orthogonal prolyl-tRNA synthetase/tRNA pairs for unnatural amino acid mutagenesis in Escherichia coli. *Proc. Natl. Acad. Sci. U. S. A.* **109**, 14841 (Sep, 2012).

111. A. Chatterjee, H. Xiao, P. Y. Yang, G. Soundararajan, P. G. Schultz, A tryptophanyl-tRNA synthetase/tRNA pair for unnatural amino acid mutagenesis in E. coli. *Angew. Chem.-Int. Edit.* **52**, 5106 (2013).

112. D. B. Goodman, G. M. Church, S. Kosuri, Causes and effects of N-terminal codon bias in bacterial genes. *Science* **342**, 475 (Oct 25, 2013).

113. K. Temme, D. Zhao, C. Voigt, Refactoring the nitrogen fixation gene cluster from Klebsiella oxytoca. *PNAS* **109**, 7085 (2012).

114. S. Kosuri *et al.*, Composability of regulatory sequences controlling transcription and translation in Escherichia coli. *Proc. Natl. Acad. Sci. U. S. A.* **110**, 14024 (Aug 20, 2013).

115. F. H. C. Crick, Codon-anticodon pairing: the wobble hypothesis. *Journal of Molecular Biology* **19**, 548 (1966).

116. S. Yokoyama, S. Nishimura, *Modified nucleosides and codon recognition*. D. Soll, U. L. RajBhandary, Eds., tRNA: Structure, biosynthesis, and function (American Society for Microbiology (ASM), Books Division, 1325 Massachusetts Ave. NW, Washington, DC 20005-4171, USA, 1995), pp. 207-223.

117. A. C. Forster, G. M. Church, Towards synthesis of a minimal cell. *Mol Syst Biol* **2**, (2006).

CHAPTER 2

Lambda Red Recombination in Escherichia coli Occurs Through a Fully Single-Stranded Intermediate

This chapter is reproduced with permission from its initial publication:

Mosberg JA[*], **Lajoie MJ[*]**, Church GM (2010) *Lambda Red Recombination in Escherichia coli Occurs Through a Fully Single-Stranded Intermediate***. GENETICS:** Vol. 186, 791-799

Research contributions:

J. Mosberg and M. Lajoie jointly devised the proposed mechanism. J. Mosberg, M. Lajoie, and G. Church planned the experiments to test the mechanistic hypothesis, and interpreted the results. M. Lajoie and J. Mosberg performed the experiments. J. Mosberg wrote the majority of the published paper, with writing and editing contributions from M. Lajoie and G. Church.

Acknowledgements:

The authors thank Farren Isaacs and Harris Wang for helpful discussions and mentorship. Francois Vigneault provided expert advice on generating single-stranded DNA. John Aach and Tara Gianoulis helped with statistical analysis. Tara Gianoulis, Srivatsan Raman, Farren Isaacs, and Uri Laserson provided valuable feedback regarding this manuscript, and Jaron Mercer assisted in conducting experiments. The *E. coli* strain SIMD90 was a gift from Donald Court, NCI-Frederick. This work was funded by the U.S. Department of Energy. M.J.L. was supported by a U.S. Department of Defense NDSEG fellowship.

Abstract

The phage Lambda-derived Red recombination system is a powerful tool for making targeted genetic changes in *Escherichia coli*, providing a simple and versatile method for generating insertion, deletion, and point mutations on chromosomal, plasmid, or BAC targets. However, despite the common use of this system, the detailed mechanism by which Lambda Red mediates double-stranded DNA recombination remains uncertain. Current mechanisms posit a recombination intermediate in which both 5′ ends of double-stranded DNA are recessed by Lambda Exonuclease, leaving behind 3′ overhangs. Here, we propose an alternative in which Lambda Exonuclease entirely degrades one strand, while leaving the other strand intact as single-stranded DNA. This single-stranded intermediate then recombines via Beta recombinase-catalyzed annealing at the replication fork. We support this by showing that single-stranded gene insertion cassettes are recombinogenic, and that these cassettes preferentially target the lagging strand during DNA replication. Furthermore, a double-stranded DNA cassette containing multiple internal mismatches shows strand-specific mutations co-segregating roughly 80% of the time. These observations are more consistent with our model than with previously proposed models. Finally, by using phosphorothioate linkages to protect the lagging-targeting strand of a double-stranded DNA cassette, we illustrate how our new mechanistic knowledge can be used to enhance Lambda Red recombination frequency. The mechanistic insights revealed by this work may facilitate further improvements to the versatility of Lambda Red recombination.

Introduction

Over the past decade, Lambda Red recombination ("recombineering") has been used as a powerful technique for making precisely defined insertions, deletions, and point mutations in

Escherichia coli, requiring as few as 35 base pairs of homology on each side of the desired alteration (*1, 2*). With this system, single-stranded DNA (ssDNA) oligonucleotides have been used to efficiently modify *E. coli* chromosomal targets (*3, 4*), BACs (*5*), and plasmids (*6*), as well as to rapidly optimize a metabolic pathway coding for the production of lycopene (*7*). Furthermore, linear double-stranded DNA (dsDNA) recombineering has been used to replace chromosomal genes (*8, 9*), to disrupt gene function (*10*), and to develop novel cloning methods (*11, 12*). Large-scale dsDNA recombineering projects include creating a library of single-gene knockout *E. coli* strains (*13*) and removing 15% of the genomic material from a single *E. coli* strain (*14*). Linear dsDNA recombineering has also been used to insert heterologous genes and entire pathways into the *E. coli* chromosome (*15, 16*) and BACs (*11, 17*), including those used for downstream applications in eukaryotes (*18, 19*). However, despite the broad use of this method, the mechanism of Lambda Red recombination has not achieved scientific consensus, particularly in the case of dsDNA recombination. A clearer understanding of the mechanism underlying this process could suggest ways to improve the functionality, ease, and versatility of Lambda Red recombination.

Three phage-derived Lambda Red proteins are necessary for carrying out dsDNA recombination: Gam, Exo, and Beta. Gam prevents the degradation of linear dsDNA by the *E. coli* RecBCD and SbcCD nucleases; Lambda Exonuclease (Exo) degrades dsDNA in a 5′ to 3′ manner, leaving single-stranded DNA in the recessed regions; and Beta binds to the single-stranded regions produced by Exo and facilitates recombination by promoting annealing to the homologous genomic target site (*20*). Current mechanisms claim that Exo binds to both 5′ ends of the dsDNA and degrades in both directions simultaneously to produce a double-stranded

region flanked on both sides by 3' overhangs (*2, 21*). However, a comprehensive explanation of how this construct ultimately recombines with the chromosome has not yet been advanced.

Initially, it was proposed that this recombination occurs via strand invasion (*22*). However, it has more recently been shown that strand invasion is unlikely to be the dominant mechanism in the absence of long regions of homology, as recombination remains highly proficient in a *recA*$^{-}$ background (*23*). Furthermore, a detailed analysis of Lambda Red recombination products showed characteristics consistent with strand annealing rather than a strand invasion model (*24*). Finally, Lambda Red dsDNA recombination has been shown to preferentially target the lagging strand during DNA replication, which suggests strand annealing rather than strand invasion (*25, 26*).

To explain these results, Court *et al.* (*27*) proposed a strand annealing model for insertional dsDNA recombination (Figure 2-1A), in which one single-stranded 3′ end anneals to its homologous target at the replication fork. The replication fork then stalls, due to the presence of a large dsDNA non-homology (*i.e.*, the insertion cassette). The stalled replication fork is ultimately rescued by the other replication fork traveling in the opposite direction around the circular bacterial chromosome. The other 3′ end of the recombinogenic DNA anneals to the homology region exposed by the second replication fork, forming a crossover structure, which is then resolved by unspecified *E. coli* enzymes (*27*).

The Court mechanism was challenged by Poteete (*25*), who showed that the dsDNA recombination of a linear Lambda phage chromosome occurs readily onto a unidirectionally-replicating plasmid, which does not have the second replication fork required by the Court mechanism (*27*). Thus, Poteete proposed an alternate mechanism (*25*), termed “replisome invasion” (Figure 2-1B), in which a 3′ overhang of the Exo-processed dsDNA first anneals to its

complementary sequence on the lagging strand of the recombination target. Subsequently, this overhang displaces the leading strand, thereby serving as the new template for leading strand synthesis. The resulting structure is resolved by an unspecified endonuclease, after which the recombinogenic DNA becomes the template for the synthesis of both new strands. In the context of recombineering using a linear dsDNA cassette, the author indicates that a second strand switching event must occur at the other end of the incoming dsDNA.

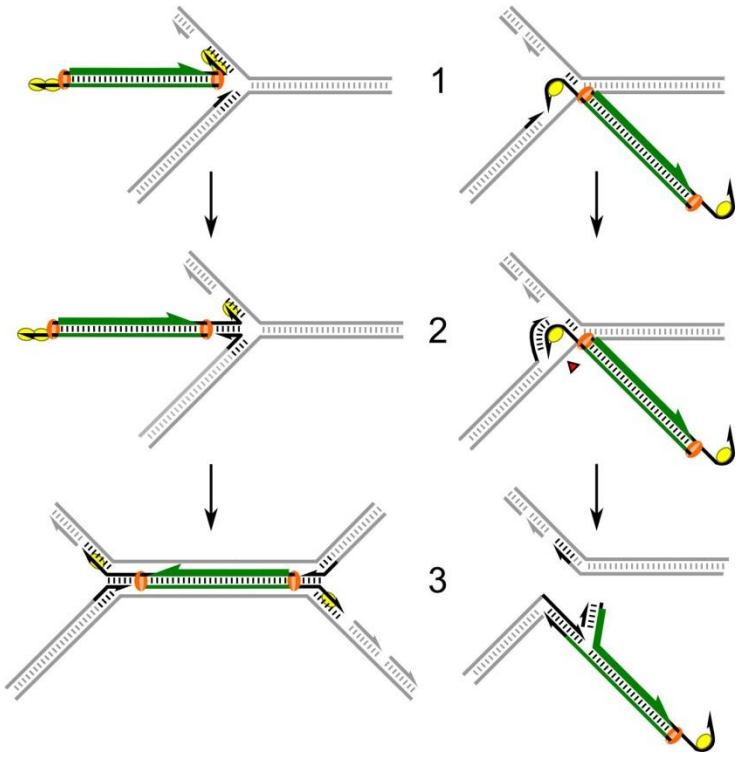


Figure 2-1. Previously proposed Lambda Red-mediated dsDNA recombination mechanisms. Heterologous dsDNA is shown in green; Exo is an orange oval, and Beta is a yellow oval. In both mechanisms the recombination intermediate is proposed to be a dsDNA core flanked on either side by 3' ssDNA overhangs. **(A)** The Court mechanism posits that 1) Beta facilitates annealing of one 3' overhang to the lagging strand of the replication fork. 2) This replication fork then stalls and backtracks so that the leading strand can template switch onto the synthetic dsDNA. The heterologous dsDNA blocks further replication from this fork. 3) Once the second replication fork reaches the stalled fork, the other 3′ end of the integration cassette is annealed to the lagging strand in the same manner as prior. Finally, the crossover junctions must be resolved by unspecified E. coli enzymes (*27*). **(B)** The Poteete mechanism suggests that 1) Beta facilitates 3' overhang annealing to the lagging strand of the replication fork, and 2) positions the invading strand to serve as the new template for leading strand synthesis. This structure is resolved by an unspecified host endonuclease, and 3) the synthetic dsDNA becomes template for both lagging and leading strand synthesis. A second template switch must then occur at the other end of the synthetic dsDNA (*25*). Both figures were adapted from the indicated references.

While Poteete's mechanism addresses some of the weaknesses of the Court mechanism, it remains largely speculative. This mechanism does not identify the endonuclease responsible for resolving the structure after the first template switching event, nor does it explain how the recombinogenic DNA and replication machinery form a new replication fork. Additionally, this template switching mechanism would have to

operate two times in a well-controlled manner, which may not be consistent with the high recombination frequencies often observed (*9*) for Lambda Red-mediated dsDNA insertion. Finally, little experimental evidence has been advanced to directly support this hypothesis.

To address the deficiencies in these mechanisms, we propose that Lambda Red dsDNA recombination proceeds via a ssDNA intermediate rather than a dsDNA core flanked by 3' overhangs (Figure 2-2). In this mechanism, Exo binds to one of the two dsDNA strands and degrades that strand completely, leaving behind full-length ssDNA. This ssDNA then anneals to its homology target at the lagging strand of the replication fork, and is incorporated as part of the newly-synthesized strand as if it were an Okazaki fragment. This process is analogous to the accepted mechanism for the Lambda Red-mediated recombination of ssDNA oligonucleotides (*27*), and therefore unifies the mechanisms for ssDNA and dsDNA recombination. Notably, our mechanism uses one replication fork for the incorporation of a full-length heterologous cassette, thereby addressing Poteete's criticism of the Court mechanism.

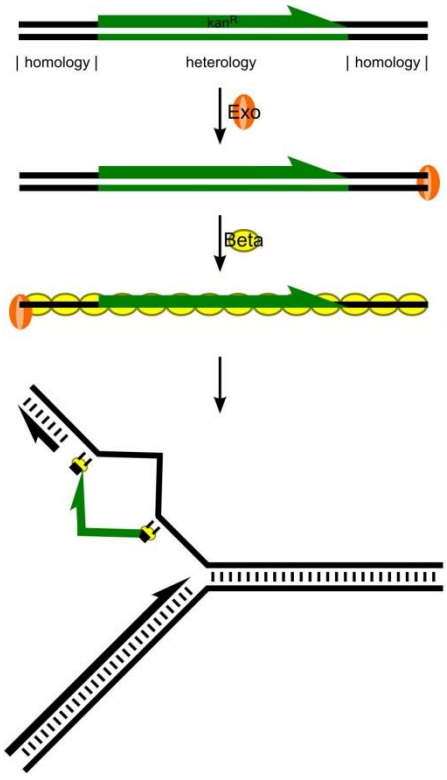


Figure 2-2. Lambda Red mediated dsDNA recombination proceeds via a ssDNA intermediate. Instead of a recombination intermediate involving dsDNA flanked by 3' ssDNA overhangs, we propose that one strand of linear dsDNA is entirely degraded by Exo (orange oval). Beta (yellow oval) then facilitates annealing to the lagging strand of the replication fork in place of an Okazaki fragment. The heterologous region does not anneal to the genomic sequence. This mechanism could account for gene replacement (as shown), or for insertions in which no genomic DNA is removed.

The degradation of an entire strand by Lambda Exo is feasible, given the highly processive nature of the enzyme (*28*). Whereas previously proposed mechanisms assume that both dsDNA ends are degraded approximately simultaneously, our hypothesis implies that some dsDNA molecules will be entirely degraded to ssDNA before a

second Exo can bind to the other end. In this manuscript, we demonstrate that single-stranded DNA is a viable recombinogenic intermediate with lagging strand bias. Furthermore, we show that genetic information from one strand of a recombinogenic dsDNA cassette co-segregates during Lambda Red-mediated recombination. These results provide strong support of our proposed mechanism.

Results

Testing the predicted ssDNA recombination intermediate: We designed a *lacZ::kanR* cassette (~1.2 kb), consisting of a kanamycin resistance gene (*kanR*) flanked by 45 bp regions homologous to the *lacZ* gene on the *E. coli* chromosome. Successful *kanR* insertion disrupted LacZ function, so proper targeting of the *lacZ::kanR* cassette could be verified by selecting on kanamycin and assaying for the inability to cleave X-Gal in order to release a blue chromophore. This dsDNA construct was generated by PCR and converted into ssDNA using a biotin capture and DNA melting protocol (*29*), as detailed in File S1. PAGE analysis confirmed the purity of the *lacZ::kanR* ssDNA construct, as no dsDNA band was readily detected. This construct was then recombined into EcNR2 (*7*). The *lacZ::kanR* ssDNA construct was found to yield $1.3 \times 10^{-5} \pm 4.5 \times 10^{-6}$ recombinants per viable cell, in comparison with $1.9 \times 10^{-4} \pm 7.5 \times 10^{-5}$ for the corresponding dsDNA construct. Both ssDNA and dsDNA gave over 99% white colonies, indicating correct targeting of the recombinogenic cassette.

This result confirms that ssDNA—the predicted intermediate for our mechanism—is recombinogenic. It is, however, 14.8-fold less recombinogenic than the corresponding dsDNA. We hypothesize that this disparity is caused by ssDNA secondary structure and/or the lack of Exo-Beta synergy. Previous work has demonstrated that ssDNA oligonucleotides longer than 90

bases and/or having secondary structure with $\Delta G < -12$ kcal/mol are likely to have substantially reduced recombination frequency (*7*). Thus, we expect secondary structure to significantly diminish the recombination frequency of this ~1.2 kb cassette. Additionally, it has previously been suggested (*30*) that Exo and Beta act synergistically, with Exo facilitating the binding of Beta to recessed regions of ssDNA. Since Exo does not readily bind to ssDNA, this synergistic action cannot occur; therefore, recombination frequency may decrease. However, even in light of these considerations, our predicted ssDNA intermediate is highly capable of recombination.

In order to confirm that the observed recombinants arose from the ssDNA rather than from dsDNA contamination, this recombination experiment was repeated in SIMD90 (*30*), a strain of *E. coli* containing Beta, but lacking Exo and Gam. In the absence of Exo and Gam, dsDNA recombination frequency should decline significantly due to increased dsDNA degradation and inefficient processing into ssDNA. In this strain, *lacZ::kanR* ssDNA demonstrated a recombination frequency of 1.8×10^{-4}, in comparison with a recombination frequency of only 8.7×10^{-7} for dsDNA (a 209-fold difference). This result indicates that the observed recombinants in EcNR2 also arose from ssDNA.

Investigating the strand bias of the recombination intermediate: We propose that the long ssDNA intermediate recombines by annealing at the replication fork in the same manner as ssDNA oligonucleotides (*27*). It has been demonstrated that lagging-targeting oligonucleotides recombine with substantially greater frequency than the corresponding leading-targeting oligonucleotides, due to the greater accessibility of the lagging strand for annealing (*31*). In order to test whether long ssDNA recombines in the same manner, we investigated whether several pairs of lagging-targeting and leading-targeting ssDNA insertion cassettes demonstrated a similar strand bias. We controlled for the effect of differential secondary structure between the two

strands by recombining three different antibiotic resistance markers into *lacZ* – kanamycin (*lacZ::kanR*), zeocin (*lacZ::zeoR*), and spectinomycin (*lacZ::specR*). Additionally, in order to demonstrate that strand bias was not caused by replichore-specific context or transcriptional direction, we constructed two additional *kanR* cassettes. To this end, *tolC::kanR* targets a gene located on the opposite replichore from *lacZ,* and *malK::kanR* targets a gene transcribed from the opposite strand of the chromosome as *lacZ.* As shown in Figure 2-3, the lagging-targeting strand was substantially more recombinogenic than the leading-targeting strand for all of the tested constructs. As previously observed for oligonucleotides (*3*), there appears to be a significant amount of locus-specific and sequence-specific variability in recombination frequency. Interestingly, a significant number of mistargeted recombinants (antibiotic-resistant colonies that retained LacZ function) were observed for both *lacZ::specR* strands (Table S2-2; discussion in File S2). Mistargeted ($LacZ^+$) colonies were not scored as recombinants, and do not affect the broader interpretation of our results. The overall results of this set of experiments clearly indicate a robust lagging strand bias, likely arising from the greater accessibility of the lagging strand during DNA replication. This

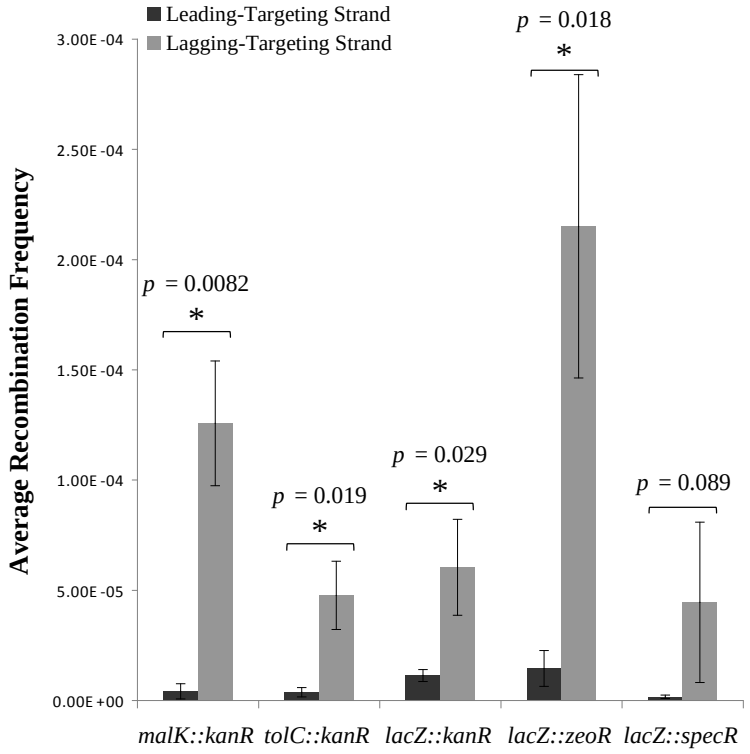


Figure 2-3. Strand bias in Lambda Red ssDNA insertion recombination. Recombination frequencies were assessed for several leading-targeting and lagging-targeting complementary ssDNA pairs. Lagging-targeting strands were found to be more recombinogenic than leading-targeting strands. An asterisk indicates $P < 0.05$.

supports our claim that long ssDNA insertion constructs recombine by annealing at the replication fork in a manner similar to ssDNA oligonucleotides.

Testing Mechanistic Predictions by Tracking Designed Mutations: The prior experiments provide strong indirect evidence supporting our proposed ssDNA annealing mechanism. In order to more directly test the predictions of this mechanism, we designed a *lacZ::kanR* dsDNA cassette with internal mismatches (Figure 2-4), which enables us to empirically determine which strand provided genetic information during recombination. This construct was generated by annealing two strands of ssDNA and purifying the resulting dsDNA by agarose gel extraction. In each of the flanking *lacZ* homology regions, this construct contains two sets of adjacent dinucleotide mismatches that differentiate the two strands. At these loci, neither strand's sequence matches the targeted chromosomal copy of *lacZ*. Thus, one can infer which strand has recombined by observing which strand-specific alleles are present.

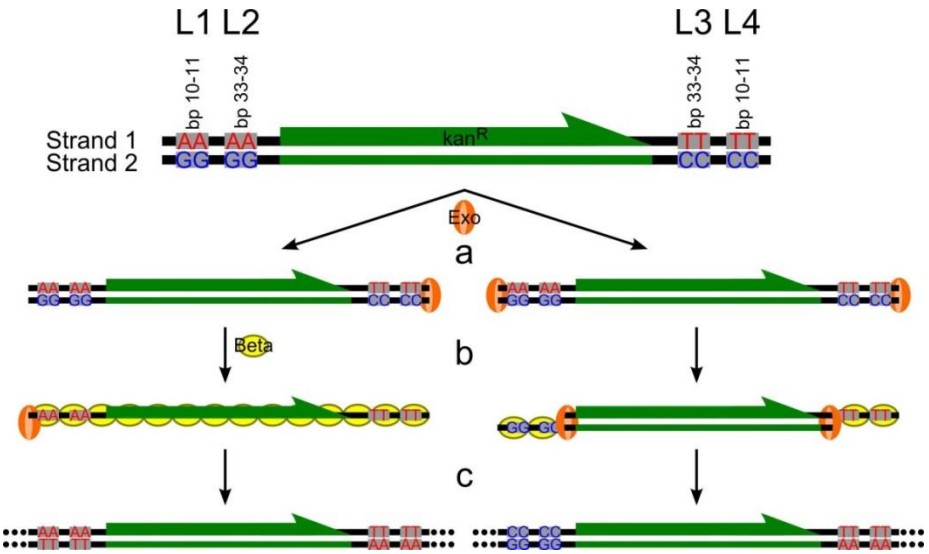


Figure 2-4. Strand-specific mismatch alleles were used to identify the strand of origin for each recombined mutation. The mismatched *lacZ::kanR* cassette contained two consecutive mismatches at two loci in both flanking homology regions. Strand 1 was the lagging-targeting strand and strand 2 was the leading-targeting strand. If Lambda Red dsDNA recombination proceeds via a ssDNA intermediate (left), a) one Exo (orange oval) binds to a dsDNA end, b) Exo fully degrades one strand while helping to load Beta (yellow oval) onto the remaining strand, and c) this strand provides all of the genetic information during recombination. This figure shows the case in which the lagging-targeting strand is recombined (coding strand genotypes: L1 = AA, L2 = AA, L3 = TT, L4 = TT), but the leading-targeting strand is also predicted to be observed (coding strand genotypes: L1 = CC, L2 = CC, L3 = GG, L4 = GG). If the Lambda Red recombination intermediate is a heterologous dsDNA core flanked by 3' ssDNA overhangs (right), a) one Exo binds to each dsDNA end, b) Exo recesses both strands while helping to load Beta onto both 3' overhangs, and c) both strands provide genetic information for each recombination. Since Exo always degrades 5'→3', the expected coding strand genotypes for the Court and Poteete mechanisms would be L1 = CC, L2 = CC, L3 = TT, L4 = TT.

Our proposed ssDNA annealing mechanism can be distinguished from the prevailing dsDNA recombination mechanisms based on the results of this experiment. Our mechanism predicts that the mutations contained on a single strand will be inherited together, and that the mutations arising from the lagging-targeting strand will be observed more frequently than those from the opposite strand. Conversely, as detailed in Figure 2-4, the previously proposed mechanisms predict that the alleles on the 3′ ends of both strands would be incorporated.

This mismatched *lacZ::kanR* cassette was transformed into EcNR2, which is deficient for mismatch repair. Recombinants were identified by plating on kanamycin, and colonies were screened using MAMA PCR (*32*) in order to identify which strand-specific mutations were inherited in each colony. Two replicates were performed, and 48 colonies were screened for each recombination (Table 2-1; detailed results in Table S2-3). The accuracy of the MAMA PCR assay was confirmed by sequencing the relevant regions of several colonies and by performing a complementary MAMA PCR assay to detect unaltered wild-type alleles at the targeted loci. In line with our predictions, we found that roughly 80% of the colonies inherited mismatch alleles from only one strand. Furthermore, of these colonies, 91% inherited mismatch alleles specifically from the lagging-targeting strand, strongly supporting our ssDNA annealing mechanism.

Table 2-1. Tracking co-segregation in mismatched dsDNA recombination

Origin of Mismatches	**Number of Recombinants Observed**
Only strand 1	68
Only strand 2	7
Split as Expected for 5' Resection	10
Split as Expected for 3' Resection	9
Ambiguous	2

Half of the remaining 20% of the colonies showed an inheritance pattern consistent with resection from both 5' ends, and the other half was consistent with resection from both 3' ends. Resection from the 5' ends is predicted by the previously proposed mechanisms, and indicates that one of these mechanisms may also operate as a disfavored process. However, Exo has not been shown to degrade dsDNA in a 3'→5' manner, even though our results imply that this occurs nearly as often as 5'→3' resection. A plausible explanation for this discrepancy is that the colonies possessing alleles from both strands have instead undergone two sequential recombination events according to our proposed mechanism. The first recombination would proceed normally, and the second recombination would involve a partially degraded complementary strand. This second recombination event would be expected to occur quite frequently – after the first recombination event, the *kanR* gene is present in the genome, providing a large region of homology to which remaining fragments of *kanR* ssDNA can anneal in subsequent rounds of replication.

Interestingly, mutations arising from loci one and four (Figure 2-4) are observed only rarely in the studied recombinants. This result suggests that a significant portion of the DNA may be undergoing slight exonuclease degradation from both the 5′ and 3′ ends, or that annealed strands are processed at the replication fork in a manner that degrades or excludes the distal ends of the recombined DNA. This is consistent with a previous observation that mutations placed on the ends of a 90 bp oligonucleotide are inherited at a substantially lower frequency than mutations placed nearer to the center of the same strand. Elucidating the basis of this phenomenon may shed more light on the detailed mechanism of Lambda Red recombination.

Nevertheless, the results from this experiment provide direct evidence that our proposed mechanism is the dominant process by which Lambda Red dsDNA recombination occurs.

Phosphorothioate Placement Alters Recombination Frequency: Leveraging our increased understanding of Lambda Red dsDNA recombination, we enhanced recombination frequency by over an order of magnitude. Since the lagging-targeting strand is the most important recombinogenic species, we reasoned that protecting this strand would improve recombination frequency. It is known that phosphorothioate bonds diminish the ability of many exonucleases to degrade DNA (*33*). In order to test whether altering phosphorothioate placement changes the resulting recombination frequencies, we made four variants of the mismatched *lacZ::kanR* dsDNA cassette, as denoted in Figure 2-5. These cassettes were recombined into EcNR2, and recombination frequencies were determined (Figure 2-5).

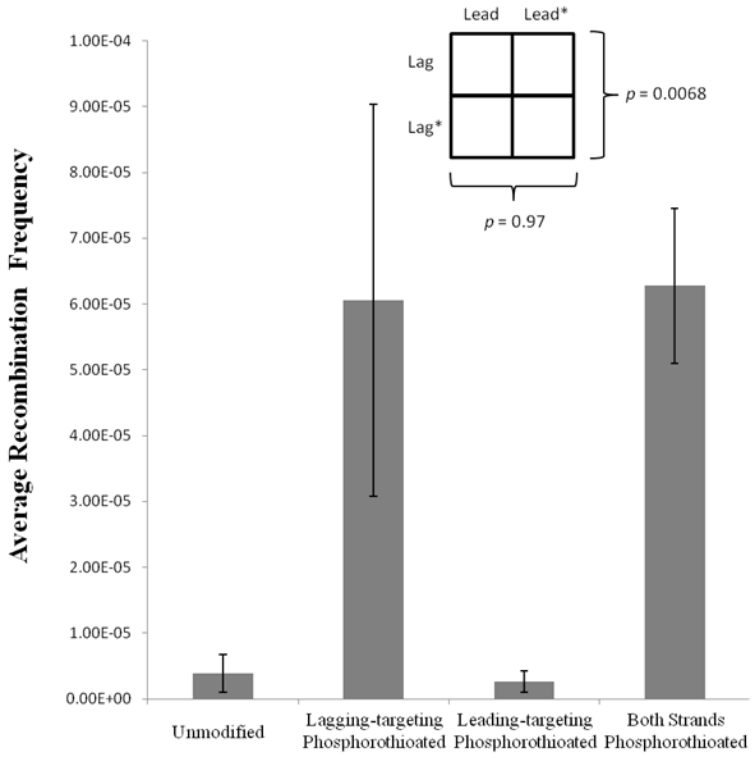


Figure 2-5. Testing the effect of strand protection on recombination frequency. Four *lacZ::kanR* cassettes were tested in order to determine whether protecting one strand has a greater effect on recombination frequency than protecting the other strand. In each case, protection was accomplished through the placement of four phosphorothioate linkages on the 5′ end of a strand. **Inset:** Analysis of variance for lagging-targeting (Lag) phosphorothioation and leading-targeting (Lead) phosphorothioation. An asterisk (*) denotes phosphorothioation. Lagging-targeting phosphorothioation was found to significantly enhance recombination frequency, whereas leading-targeting phosphorothioation did not affect recombination frequency.

These results show that protecting the lagging-targeting strand with phosphorothioate bonds increases the frequency of

dsDNA recombination, whereas protecting the leading-targeting strand has no effect. This further supports our proposed mechanism, since alternating which of the two strands is protected by phosphorothioates would not be expected to have differential effects if resection occurred from both 5′ ends. Additionally, our results unexpectedly show that lagging-targeting strand protection and dual protection yield approximately equivalent recombination frequencies. This suggests that phosphorothioation does not significantly inhibit *in vivo* Exo degradation, as dual protection would prohibit processing by Exo if this were the case. Instead, it is likely that placing phosphorothioates on the lagging-targeting strand protects it from host exonuclease degradation after Exo processing. This result demonstrates how our improved mechanistic knowledge of Lambda Red recombination can facilitate rational improvements of the process.

Discussion

This work provides strong empirical support for the proposed mechanism that Lambda Red dsDNA recombination operates through a full-length ssDNA intermediate. This mechanism appears to be the dominant means of Lambda Red dsDNA recombination, although other mechanisms may still occur as minor processes. Notably, a replisome invasion mechanism (*25*) involving a fully single-stranded intermediate is not directly refuted by our work, although a strand annealing model is favorable due to its well-precedented (*24, 27*) and parsimonious nature.

While our mechanism has not previously been postulated as the manner by which the Lambda Red system recombines large dsDNA segments, it is consistent with numerous results observed by other groups. By annealing two staggered oligonucleotides, Yu *et al.* previously generated a 106 bp construct consisting of a dsDNA core flanked by 3′ overhangs – the

recombination intermediate predicted by the canonical model of Lambda Red dsDNA recombination (*34*). As expected, recombination of this construct did not depend on the presence of Exo; however, even in the presence of Exo, the recombination frequency was roughly 4000-fold lower than that of its corresponding dsDNA. Given that the construct with 3′ overhangs is postulated to be a downstream intermediate of this dsDNA, this result casts doubt upon the claim that the tested construct is indeed the predominant recombination intermediate. However, this result is explained by our proposed mechanism – only the intact dsDNA can generate the full-length ssDNA needed to undergo recombination, as neither individual strand of the construct containing 3′ overhangs is sufficient for recombination (*34*). We suggest that this 3′ overhang construct recombines by a separate and disfavored process. This is supported by the fact that this proposed recombination intermediate had no greater recombination frequency than the corresponding structure with 5′ (rather than 3′) overhangs. It is unlikely that either of these structures represents the predominant intermediate of dsDNA recombination.

Muyrers *et al.* (*35*) have also provided evidence contrary to a dsDNA recombination intermediate containing 3′ overhangs. The authors created a dsDNA construct in which phosphorothioate linkages placed between an antibiotic resistance gene and its flanking genome homology regions were used to prevent exonuclease degradation beyond these homology regions. Two 5′-to-3′ exonucleases other than Exo were then used *in vitro* to resect the 5' ends of this construct, in order to generate the putative intermediate for dsDNA recombination. However, it was found that none of the tested resection conditions could produce a construct that would recombine in the absence of Exo. In contrast, the predicted intermediate in our proposed mechanism is highly recombinogenic – even when prepared *in vitro*.

Additionally, other prior work supports our proposed mechanism by reinforcing the processive nature of Exo. Hill *et al*. showed that non-replicating Lambda phage in *E. coli* is capable of converting linear dsDNA into ssDNA, creating single-stranded regions that span more than 1.4 kb (*36*). They also demonstrated that *exo* is sufficient for generating these regions of ssDNA, which are similar in length to the ~1.2 kb constructs used in this experiment. An additional implication of this result is that a single-stranded intermediate is also present during crosses involving an intact Lambda chromosome. These results suggest that our proposed mechanism may apply for natural Lambda Red recombination between phage and bacterial chromosomes. By extension, this model may also describe crosses between the phage chromosome and a plasmid (*25*), as plasmids present an accessible lagging strand at the replication fork in the same manner as the bacterial chromosome.

The results of Lim *et al.* (*26*) further reinforce that Exo generates long strands of ssDNA. These researchers created a dsDNA construct in which two antibiotic resistance genes were attached via a genome homology region and flanked with two additional regions of genome homology. Using this cassette, only about 10% of recombinants incorporated both resistance genes, while a majority of recombinants incorporated only one of the two. This implies that a majority of recombination events involved the central homology region, which is roughly 1 kb away from either end of the dsDNA construct. Given that strand annealing requires exposed ssDNA, this result further suggests that Exo can be substantially processive *in vivo*, degrading large stretches of DNA rather than short flanking segments. Indeed, limits to the processivity of Exo could explain why recombination frequency declines substantially for increasing dsDNA insertion sizes, but not for increasing chromosomal deletion sizes (*37*).

Finally, while this manuscript was in revision, Maresca *et al.* (*37*) published complementary results, in which strand-specific 5' phosphorylation and phosphorothioation were used to bias Exo degradation to each strand of a selectable cassette. For recombination events following both *in vitro* and *in vivo* digestion, the authors observed a lagging-targeting strand bias. Building upon these observations, the authors identified ssDNA as a recombinogenic species, and proposed a mechanism consistent with the one advanced in this manuscript. These results provide substantial validation of our model. Our experiment involving a mismatched dsDNA cassette extends this work by showing that information from a single strand co-segregates during Lambda Red mediated dsDNA recombination. More importantly than simply showing a lagging-targeting strand bias, this experiment provides direct evidence of a single-stranded intermediate in Lambda Red dsDNA recombination.

Our proposed mechanism may also describe other recombineering processes mediated by Lambda Red. One example is gap repair, in which linearized plasmid DNA is used to capture chromosomal DNA (*11, 27*). Notably, while a detailed mechanism has not yet been advanced for Lambda Red-facilitated gap repair, our model involving a single-stranded intermediate provides a plausible explanation. Given a full-length ssDNA intermediate, the linearized plasmid would anneal to the chromosomal target with its homology regions facing one another. The 3' end homology would then be elongated in the direction of the 5' end homology, thereby introducing the chromosomal DNA of interest into the plasmid. The linear single-stranded plasmid would be circularized by ligase in the same manner as Okazaki fragment joining. The circular ssDNA would then be liberated from the chromosome, possibly during chromosomal replication. Residual ssDNA from the other strand of the linearized plasmid may be necessary to prime replication in order to convert the circular single-stranded plasmid into double-stranded DNA.

Notably, this mechanism accounts for the gap repair of large (> 80 kb) genomic sequences (*38*), since the two homology regions could anneal with multiple Okazaki fragments between them. These fragments would then be joined by the natural lagging strand replication mechanism.

In conclusion, a large body of evidence from our current work and from previously published studies supports our claim that the predominant mechanism for Lambda Red dsDNA recombination involves the annealing of a full-length ssDNA intermediate to the lagging strand of the replication fork. However, it is possible that previously suggested mechanisms involving the resection of both 5′ ends still operate as a minor process. The mismatched dsDNA approach described in this work may be a powerful platform to further explore the extent to which any such minor recombination mechanisms may operate.

The mechanism of Lambda Red recombination has long been a matter of debate (*21*). This work posits and supports a novel mechanism, which may reveal improved recombination parameters that increase the frequency and robustness of recombineering. Just as the mechanistic understanding of Red-mediated oligonucleotide recombination facilitated its optimization and use in novel and powerful applications (*7*), similar innovations may provide for transformative applications of Lambda Red dsDNA recombination.

Materials and Methods

The preparation of the various DNA constructs used in this study is detailed in the Supplemental Materials and Methods section (File S1). These DNA constructs were recombined into EcNR2 cells {*E. coli* MG1655 Δ(*ybhB-bioAB*)::[λcI857 Δ(*cro-ea59*)::*tetR-bla*] Δ*mutS*::*cat*} in a similar manner as previously described (*7*). Briefly, cells were grown in a rotator drum at 32 °C in LB-min media (10 g tryptone, 5 g yeast extract, 5 g sodium chloride per 1 L water) until

they reached an OD_{600} of 0.4 – 0.6. At this time, the expression of the Lambda Red proteins was induced by vigorously shaking the cells in a 42 °C water bath for 15 minutes. Cells were then chilled on ice, washed twice with deionized water, and resuspended in 50 µL of deionized water containing the desired DNA construct. For the experiment investigating strand bias, 20 ng of DNA was recombined. For all other experiments, 50 ng was used. The DNA was then introduced into the cells via electroporation (BioRad Gene Pulser™; 0.1 cm cuvette, 1.78 kV, 25 µF, 200 Ω). After electroporation, cells were recovered in 3 mL LB-min media for 3 hours in a rotator drum at 32 °C.

Recombinants were identified by plating 50 µL or 1 mL (concentrated to 50 µL) of undiluted recovery culture on selective media (LB-min with 30 µg/mL kanamycin sulfate, 95 µg/mL spectinomycin, or 10 µg/mL Zeocin™). The total viable cell count was determined by plating 50 µL of a 10^{-4} dilution of the recovery culture (in LB-min) onto LB-min + 20 µg/mL chloramphenicol plates. For experiments involving *lacZ* gene disruption, the plates also contained Fisher ChromoMax™ X-Gal/IPTG solution at the manufacturer's recommended concentration. Recombination frequencies were determined by dividing the extrapolated number of recombinants by the total viable cell count. All experiments assessing recombination frequency were performed in triplicate, except the series of recombinations in which phosphorothioate placement was altered – these were performed in duplicate. The recombination frequencies determined for each replicate were averaged and the error of the mean was taken to be $\sigma_{\bar{x}} = \frac{\sigma_x}{\sqrt{N}}$. We tested our hypothesis that lagging strand recombination frequency is higher than leading strand recombination frequency by using a one-tailed *t*-test assuming unequal variances. We used a two-way analysis of variance test with two replicates to assess significance of the phosphorothioate protection experiment. Both statistical analyses were calculated with

default parameters by MATLAB. Following the *lacZ::kanR* mismatched dsDNA recombinations, mismatch amplification mutation assay (MAMA) PCR (see File S1 for detailed description) was used for genotypic analysis. A complete list of primers used in the study can be found in the supplemental material (Table S2-1).

Supplemental material

Supplemental material for CHAPTER 2 can be found in APPENDIX A or at <http://www.genetics.org/content/suppl/2010/09/02/genetics.110.120782.DC1/120782_SI.pdf>.

References

1. L. Thomason *et al.*, Recombineering: genetic engineering in bacteria using homologous recombination. *Curr Protoc Mol Biol* **Chapter 1**, Unit 1 16 (Apr, 2007).

2. S. K. Sharan, L. C. Thomason, S. G. Kuznetsov, D. L. Court, Recombineering: a homologous recombination-based method of genetic engineering. *Nat Protoc* **4**, 206 (2009).

3. H. M. Ellis, D. Yu, T. DiTizio, D. L. Court, High efficiency mutagenesis, repair, and engineering of chromosomal DNA using single-stranded oligonucleotides. *Proc Natl Acad Sci U S A* **98**, 6742 (Jun 5, 2001).

4. N. Costantino, D. L. Court, Enhanced levels of lambda Red-mediated recombinants in mismatch repair mutants. *Proc Natl Acad Sci U S A* **100**, 15748 (Dec 23, 2003).

5. S. Swaminathan *et al.*, Rapid engineering of bacterial artificial chromosomes using oligonucleotides. *Genesis* **29**, 14 (Jan, 2001).

6. L. C. Thomason, N. Costantino, D. V. Shaw, D. L. Court, Multicopy plasmid modification with phage lambda Red recombineering. *Plasmid* **58**, 148 (Sep, 2007).

7. H. H. Wang *et al.*, Programming cells by multiplex genome engineering and accelerated evolution. *Nature* **460**, 894 (Aug, 2009).

8. K. C. Murphy, Use of bacteriophage lambda recombination functions to promote gene replacement in Escherichia coli. *J Bacteriol* **180**, 2063 (Apr, 1998).

9. K. C. Murphy, K. G. Campellone, A. R. Poteete, PCR-mediated gene replacement in Escherichia coli. *Gene* **246**, 321 (Apr 4, 2000).

10. K. A. Datsenko, B. L. Wanner, One-step inactivation of chromosomal genes in Escherichia coli K-12 using PCR products. *Proc Natl Acad Sci U S A* **97**, 6640 (Jun 6, 2000).

11. E. C. Lee *et al.*, A highly efficient Escherichia coli-based chromosome engineering system adapted for recombinogenic targeting and subcloning of BAC DNA. *Genomics* **73**, 56 (Apr 1, 2001).

12. M. Z. Li, S. J. Elledge, MAGIC, an in vivo genetic method for the rapid construction of recombinant DNA molecules. *Nat Genet* **37**, 311 (Mar, 2005).

13. T. Baba *et al.*, Construction of Escherichia coli K-12 in-frame, single-gene knockout mutants: the Keio collection. *Mol Syst Biol* **2**, 2006 0008 (2006).

14. G. Posfai *et al.*, Emergent properties of reduced-genome Escherichia coli. *Science* **312**, 1044 (May 19, 2006).

15. Y. Zhang, F. Buchholz, J. P. Muyrers, A. F. Stewart, A new logic for DNA engineering using recombination in Escherichia coli. *Nat Genet* **20**, 123 (Oct, 1998).

16. Y. Wang, B. A. Pfeifer, 6-deoxyerythronolide B production through chromosomal localization of the deoxyerythronolide B synthase genes in E. coli. *Metab Eng* **10**, 33 (Jan, 2008).

17. S. Warming, N. Costantino, D. L. Court, N. A. Jenkins, N. G. Copeland, Simple and highly efficient BAC recombineering using galK selection. *Nucleic Acids Res* **33**, e36 (2005).

18. M. K. Chaveroche, J. M. Ghigo, C. d'Enfert, A rapid method for efficient gene replacement in the filamentous fungus Aspergillus nidulans. *Nucleic Acids Res* **28**, E97 (Nov 15, 2000).

19. J. Bouvier, J. G. Cheng, Recombineering-based procedure for creating Cre/loxP conditional knockouts in the mouse. *Curr Protoc Mol Biol* **Chapter 23**, Unit 23 13 (Jan, 2009).

20. J. A. Sawitzke *et al.*, Recombineering: in vivo genetic engineering in E. coli, S. enterica, and beyond. *Methods Enzymol* **421**, 171 (2007).

21. A. K. Szczepanska, Bacteriophage-encoded functions engaged in initiation of homologous recombination events. *Crit Rev Microbiol* **35**, 197 (2009).

22. D. S. Thaler, M. M. Stahl, F. W. Stahl, Double-chain-cut sites are recombination hotspots in the Red pathway of phage lambda. *J Mol Biol* **195**, 75 (May 5, 1987).

23. D. Yu *et al.*, An efficient recombination system for chromosome engineering in Escherichia coli. *Proc Natl Acad Sci U S A* **97**, 5978 (May 23, 2000).

24. M. M. Stahl *et al.*, Annealing vs. invasion in phage lambda recombination. *Genetics* **147**, 961 (Nov, 1997).

25. A. R. Poteete, Involvement of DNA replication in phage lambda Red-mediated homologous recombination. *Mol Microbiol* **68**, 66 (Apr, 2008).

26. S. I. Lim, B. E. Min, G. Y. Jung, Lagging strand-biased initiation of red recombination by linear double-stranded DNAs. *J Mol Biol* **384**, 1098 (Dec 31, 2008).

27. D. L. Court, J. A. Sawitzke, L. C. Thomason, Genetic engineering using homologous recombination. *Annu. Rev. Genet.* **36**, 361 (2002).

28. K. Subramanian, W. Rutvisuttinunt, W. Scott, R. S. Myers, The enzymatic basis of processivity in lambda exonuclease. *Nucleic Acids Res* **31**, 1585 (Mar 15, 2003).

29. E. Pound, J. R. Ashton, H. A. Becerril, A. T. Woolley, Polymerase chain reaction based scaffold preparation for the production of thin, branched DNA origami nanostructures of arbitrary sizes. *Nano Lett* **9**, 4302 (Dec, 2009).

30. S. Datta, N. Costantino, X. Zhou, D. L. Court, Identification and analysis of recombineering functions from Gram-negative and Gram-positive bacteria and their phages. *Proc Natl Acad Sci U S A* **105**, 1626 (Feb 5, 2008).

31. X. T. Li *et al.*, Identification of factors influencing strand bias in oligonucleotide-mediated recombination in Escherichia coli. *Nucleic Acids Res* **31**, 6674 (Nov 15, 2003).

32. Y. Z. Qiang *et al.*, Use of a rapid mismatch PCR method to detect gyrA and parC mutations in ciprofloxacin-resistant clinical isolates of Escherichia coli. *J Antimicrob Chemother* **49**, 549 (Mar, 2002).

33. X. P. Liu, J. H. Liu, The terminal 5' phosphate and proximate phosphorothioate promote ligation-independent cloning. *Protein Sci* **19**, 967 (May, 2010).

34. D. Yu, J. A. Sawitzke, H. Ellis, D. L. Court, Recombineering with overlapping single-stranded DNA oligonucleotides: testing a recombination intermediate. *Proc Natl Acad Sci U S A* **100**, 7207 (Jun 10, 2003).

35. J. P. Muyrers, Y. Zhang, F. Buchholz, A. F. Stewart, RecE/RecT and Redalpha/Redbeta initiate double-stranded break repair by specifically interacting with their respective partners. *Genes Dev* **14**, 1971 (Aug 1, 2000).

36. S. A. Hill, M. M. Stahl, F. W. Stahl, Single-strand DNA intermediates in phage lambda's Red recombination pathway. *Proc Natl Acad Sci U S A* **94**, 2951 (Apr 1, 1997).

37. M. Maresca *et al.*, Single-stranded heteroduplex intermediates in lambda Red homologous recombination. *BMC Mol Biol* **11**, 54 (Jul 29, 2010).

38. Y. Zhang, J. P. Muyrers, G. Testa, A. F. Stewart, DNA cloning by homologous recombination in Escherichia coli. *Nat Biotechnol* **18**, 1314 (Dec, 2000).

CHAPTER 3

Manipulating Replisome Dynamics and DNA Exonucleases to Enhance Lambda Red-Mediated Multiplex Genome Engineering

This chapter is adapted from portions of the following published papers:

Mosberg JA[*], Gregg CJ[*], Lajoie MJ[*], Wang HH, Church GM (2012) *Improving Lambda Red Genome Engineering in Escherichia coli via Rational Removal of Endogenous Nucleases*. **PLoS ONE** 7(9): e44638. doi:10.1371/journal.pone.0044638

Lajoie MJ[*], Gregg CJ[*], Mosberg JA[*], Washington GC, Church GM (2012) *Manipulating Replisome Dynamics to Enhance Lambda Red-Mediated Multiplex Genome Engineering*. **NAR**; doi: 10.1093/nar/gks751

Research contributions:

M. Lajoie and J. Mosberg came up with the idea for using nuclease removal and replisome manipulation to improve MAGE performance. M. Lajoie, J. Mosberg, C. Gregg, and G. Church designed the experiments and interpreted their results. M. Lajoie, J. Mosberg, and C. Gregg performed the experiments. M. Lajoie and J. Mosberg wrote a majority of these published papers, with additional writing and editing contributions from C. Gregg and G. Church.

Acknowledgements:

The authors thank John Aach, Harris Wang, Farren Isaacs, and Nikolai Eroshenko for helpful discussions, and Sara Vassallo and Gabriel Washington for technical assistance. This work was supported by the Department of Energy Genomes to Life Center [Grant number DE-FG02-02ER63445] and by a U.S. Department of Defense NDSEG Fellowship to M.J.L. The National Institutes of Health [grant number P50 HG005550] partly supported G.C.W.

Abstract

The bacteriophage λ Red recombination system is capable of efficiently introducing targeted genetic changes in *Escherichia coli*. Previously, we utilized this system to enable multiplex automatable genome engineering (MAGE) for pathway optimization (*1*). Additionally, we demonstrated that desired mutations could be enriched by co-selection with a nearby selectable mutation (Co-Selection MAGE, CoS-MAGE) (*2, 3*). In this chapter, we have demonstrated that both synthetic oligonucleotides and accessible ssDNA targets on the lagging strand of the replication fork are limiting factors for MAGE. Based on these mechanisms, we have engineered strains that exhibit improved genome engineering characteristics. Removing a set of five exonucleases (RecJ, ExoI, ExoVII, ExoX, and λ Exo) reduces MAGE oligonucleotide degradation, and disrupting the interaction between primase and helicase increases Okazaki fragment (OF) length and accessibility at the replication fork due to less frequent primer synthesis. By combining these strain improvements, we engineered a strain which displayed 111% more alleles converted per clone, 527% more clones with five or more allele conversions, and 71% fewer clones with zero allele conversions in one cycle of 10-plex CoS-MAGE compared to a standard recombineering strain (EcNR2). These improvements will facilitate ambitious genome engineering projects by minimizing dependence on time-consuming clonal isolation and screening.

Introduction

High throughput genome engineering requires the ability to cheaply and efficiently generate exact genomic DNA sequences. In this way, *de novo* genome synthesis (*4*, *5*) is an attractive approach for generating designer organisms. However, the incomplete understanding of genome structure and function poses a significant risk of designing non-viable genomes. Therefore, it is essential to test many designs. For example, a single nucleotide DNA synthesis error in the completely *de novo* synthesized *M. mycoides* chromosome caused a frameshift in *dnaA* that prevented the transplanted genome from surviving (*5*). As *de novo* synthesis becomes commonly used for creating genomes with novel or altered functionalities, the risk of generating non-viable genomes will increase. Multiplex Automatable Genome Engineering (MAGE) is a powerful alternative strategy for engineering genomes *in vivo*. MAGE simultaneously introduces several synthesized DNA oligonucleotides (oligos), resulting in the efficient modification of the *Escherichia coli* chromosome. This technique relies on phage λ Redβ recombinase, which binds to ssDNA oligos, protecting them from ssDNA exonucleases, and facilitating their annealing to the lagging strand of the replication fork (*6*). This highly efficient process generates a diverse heterogenic population, which converges toward a fully modified isogenic population after many cycles of recombination with non-degenerate oligo pools. Generating a heterogenic population has been harnessed for directed evolution of biosynthetic pathways (*1*) and extensive cycling toward isogenic populations has been used to remove all 314 UAG stop codons in subsets across 32 *E. coli* strains (*7*). By integrating evolution with engineering, MAGE combinatorially explores a broad pool of viable and non-viable mutations. Since MAGE edits the genome *in vivo*, non-viable mutations never accumulate in the population. Yet, while this attribute of *in vivo*

genome engineering enables increasingly ambitious genome designs, the ability of MAGE to efficiently generate those designs is often a limiting factor.

Several advances have already enhanced λ Red-mediated recombination from its initial ~0.2% singleplex allele replacement (AR) frequency up to ~30% (*1*). Thus far, the predominant approach for improving Redβ-mediated AR has been to optimize oligo design. Such advances include targeting oligos to the lagging strand of the replication fork (*8*), evading mismatch repair using modified nucleotides (*9*), minimizing oligo secondary structure and optimizing homology lengths (*1*), blocking oligo degradation with 5' phosphorothioate bonds (*1*), and avoiding sequences with high degrees of off-target homology elsewhere in the genome (*7*). Additionally, removing the mismatch repair protein MutS to avoid reversion of mutated alleles (*10*) was a key innovation, but little other strain engineering was reported until recently. Such strain engineering could significantly augment the power of MAGE.

Recently, a new strategy (co-selection MAGE, or CoS-MAGE) has been developed to engineer highly modified cells. This strategy uses an oligo that repairs a broken selectable marker (*e.g.*, antibiotic resistance gene) to enhance AR frequency of nearby non-selectable alleles (*3, 11*). CoS-MAGE enhances the average multiplex allele replacement frequency approximately 4-fold by selecting for cells that take up MAGE oligos and that have a permissive replication fork in the desired region of the genome (*3*). Additionally, this approach selects against daughter cells with unaltered genomes, as it removes the population that does not revert the selectable allele (Figure 3-1).

Despite the increased efficacy of CoS-MAGE, we hypothesized that intracellular MAGE oligonucleotides and accessible ssDNA on the lagging strand of the replication fork are both limiting factors for multiplex allele replacement. Therefore, we engineered a strain to overcome

these limitations. Many lines of evidence suggest that endogenous nucleases limit recombination. As discussed previously, using phosphorothioate bonds to protect oligonucleotides (*1*) and dsDNA cassettes (*12, 13*) has been shown to improve recombination frequency, suggesting that endogenous nuclease degradation antagonizes recombination. This is bolstered by the recent observation that knocking out four potent ssDNA exonucleases improves singleplex oligonucleotide recombination frequency when low concentrations of oligos are used (*14*). Furthermore, it has been shown that mutations located near the ends of an oligonucleotide (*15*) or dsDNA cassette (*12*) are inherited less often than mutations located closer to the interior of the

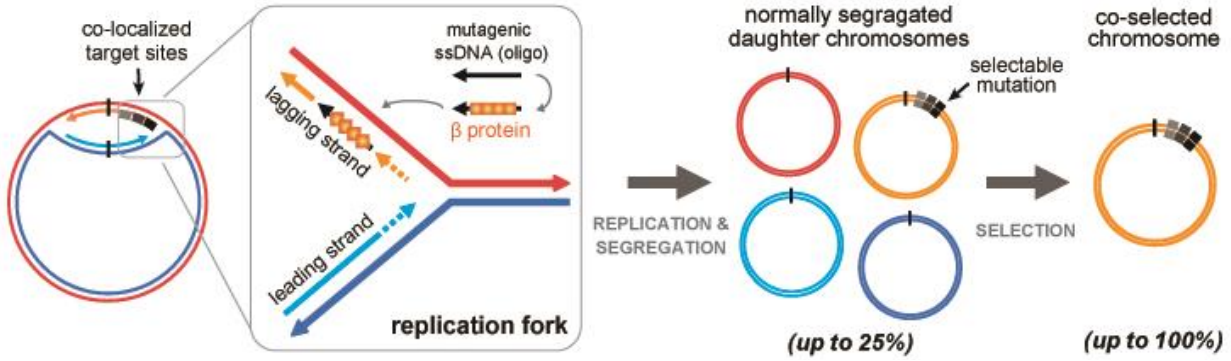


Figure 3-1. AR optimization via CoS-MAGE. The dividing chromosome is schematized, with integration of a MAGE oligo into the genome at a replication fork [adapted from Costantino and Court (*37*)]. An oligo electroporated into the cell is bound by multiple copies of the λ bacteriophage β recombinase and anneals to the lagging strand during DNA replication. When multiple oligos are incorporated into nearby sites (black and gray rectangles), they tend to co-segregate and are often inherited by the same daughter cell. Co-selection facilitates the removal of unmodified daughter chromosomes from the population, increasing AR frequency in the co-selected population. This figure is from Carr et al. (*3*).

oligo or cassette. This further implies degradation of oligonucleotides and dsDNA, and suggests that this nuclease processing prevents the inheritance of mutations along the full length of a cassette. We reasoned that by inactivating certain endogenous nucleases such as the potent ExoI, ExoVII, ExoX, RecJ, and Redα exonucleases, we could improve recombination frequency and preserve mutations encoded at ends of synthetic DNA.

In parallel, the fact that CoS-MAGE is most effective for oligos targeted in close proximity to the selectable marker suggests that replication fork position and accessibility are limiting factors in λ Red-mediated recombination (*3*). Thus, we reasoned that we could improve AR frequencies by manipulating replication fork dynamics to increase the amount of ssDNA on the lagging strand of the replication fork. Since Okazaki Fragment (OF) size can be modulated by the frequency of OF primer synthesis by DnaG primase (*16*), we hypothesized that attenuating the interaction between DnaG primase and the replisome would increase the amount of accessible ssDNA on the lagging strand of the replication fork and enhance multiplex AR frequencies (Figure 3-2). Tougu *et al.* (*17*) have reported *E. coli* primase variants with impaired helicase binding, resulting in less-frequent OF initiation, but normal replication fork rate, priming efficiency, and primer utilization during *in vitro* replication. These variants, K580A and Q576A, resulted in *in vitro* OFs that were approximately 1.5- and 8-fold longer (respectively) than those initiated by wild-type DnaG (*18*). These strains were therefore chosen to explore whether increasing accessible ssDNA on the lagging strand can improve multiplex AR frequency.

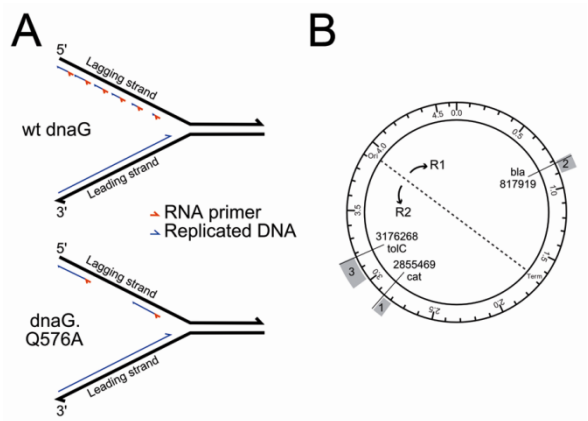


Figure 3-2. Effect of *dnaG* attenuation on replication fork dynamics. **(A)** Schematic showing the replication fork in *E. coli*, including the leading and lagging strands undergoing DNA synthesis. DnaG synthesizes RNA primers (red) onto the lagging template strand, which in turn initiate Okazaki fragment synthesis (blue) by PolIII. Compared to wt DnaG primase, the variants tested in this study have lower affinities for DnaB helicase (*17*). Since the DnaG-DnaB interaction is necessary for primase function, primer synthesis occurs less frequently, thereby exposing larger regions of ssDNA on the lagging template strand (*18*). **(B)** A schematic representing the *E. coli* MG1655 genome with the origin (*oriC*) and terminus (*T*) of replication indicated, splitting the genome into Replichore 1 and Replichore 2. Each oligo set converts 10 UAG codons to UAA codons within the genomic regions indicated in gray. Co-selection marker positions are denoted by radial lines.

In this work, we demonstrate that intracellular MAGE oligonucleotides and accessible ssDNA on the lagging strand of the

replication fork are both limiting factors for multiplex allele replacement, and that inactivating nucleases and disrupting the interaction between DnaG primase and DnaB helicase significantly improves multiplex allele replacement frequencies. We further describe the creation of an optimized strain for CoS-MAGE, which combines approaches to increase intracellular oligo concentration and to expose accessible ssDNA on the lagging strand of the replication fork. This strain demonstrates greatly improved CoS-MAGE performance, and provides a foundation for genome engineering projects of a much more ambitious scope.

Results

Nuclease Knockouts Improve MAGE Performance: It has previously been shown (*14*) that removing the four potent *E. coli* ssDNA exonucleases (ExoI, ExoVII, ExoX, and recJ) improves singleplex recombination frequency, but only when low concentrations of oligo are used. Since oligonucleotide concentration can easily be increased, this has little practical benefit. However, nuclease removal may provide a greater benefit when multiple oligonucleotides are recombined simultaneously, as in MAGE (*1*). Previous results (*3*) have shown that the recombination frequency of a given oligo is directly proportional to the mole fraction of that oligo in a complex mixture, even when the oligo is present at concentrations that would be saturating for singleplex oligo recombination. We hypothesize that this apparent competition between oligonucleotides is due to a limited number of oligos entering each cell during electroporation. Thus, if several oligos are simultaneously co-electroporated, the resulting intracellular concentration of any given oligo will be low. Presynaptic (*i.e.*, prior to incorporation) nuclease degradation may therefore have a large negative impact on recombination frequency in MAGE.

To investigate this, we compared the MAGE performance of EcNR2, EcNR2.*xseA*$^-$, and Nuc5$^-$, which is ExoI$^-$, ExoVII$^-$, ExoX$^-$, recJ$^-$, and λ Exo$^-$. In addition to the four potent exonucleases described above, λ Exo was also inactivated in this strain because it has been shown (*19*) to have trace activity against ssDNA and is not required for oligo recombination. CoS-MAGE (*3*) was used in these experiments, so as to determine whether the nuclease knockout strains are able to improve upon the current best practices for MAGE. Three sets of recombineering oligos (designed in (*7*) to convert UAG codons to UAA and renamed herein for clarity as Sets 1-3) were used in order to control for potential oligo-, allele-, region-, and replichore-specific effects (Figure 3-2B) (*7*). Each of the three oligo sets was paired with a co-selection oligo which restored the function of a nearby mutated antibiotic resistance gene (*cat* for Set 1, *bla* for Set 2, and *tolC* for Set 3), thereby selecting for high recombination frequency in the vicinity of the targeted loci. All recombineering oligos had two PT bonds on each end, as was previously optimized for MAGE (*7*). Targeted loci were screened by mascPCR (*7*) to determine which alleles were converted in a given clone.

For all three recombineering oligo sets (Set 1 in Figure 3-3A, Set 2 in Figure 3-3B, and Set 3 in Figure 3-3C), Nuc5$^-$ significantly outperforms EcNR2 (*** $P < 0.0001$, *** $P < 0.0001$, * $P = 0.002$, respectively). An average of 46% more alleles are converted per clone in Nuc5$^-$, and the frequency of clones with 5 or more conversions is increased by 200%. Furthermore, Nuc5$^-$ reduces the frequency of clones with no conversions by 35%.

The EcNR2.*xseA*$^-$ strain appears to have properties intermediate between those of EcNR2 and Nuc5$^-$. Although EcNR2.*xseA*$^-$ exhibits a statistically significant increase in MAGE performance with Set 1 (1.47 ± 0.13) compared to EcNR2 (0.96 ± 0.07, ** $P = 0.0001$), this strain's performance with Sets 2 and 3 was not statistically different from EcNR2 ($P = 0.7$ and

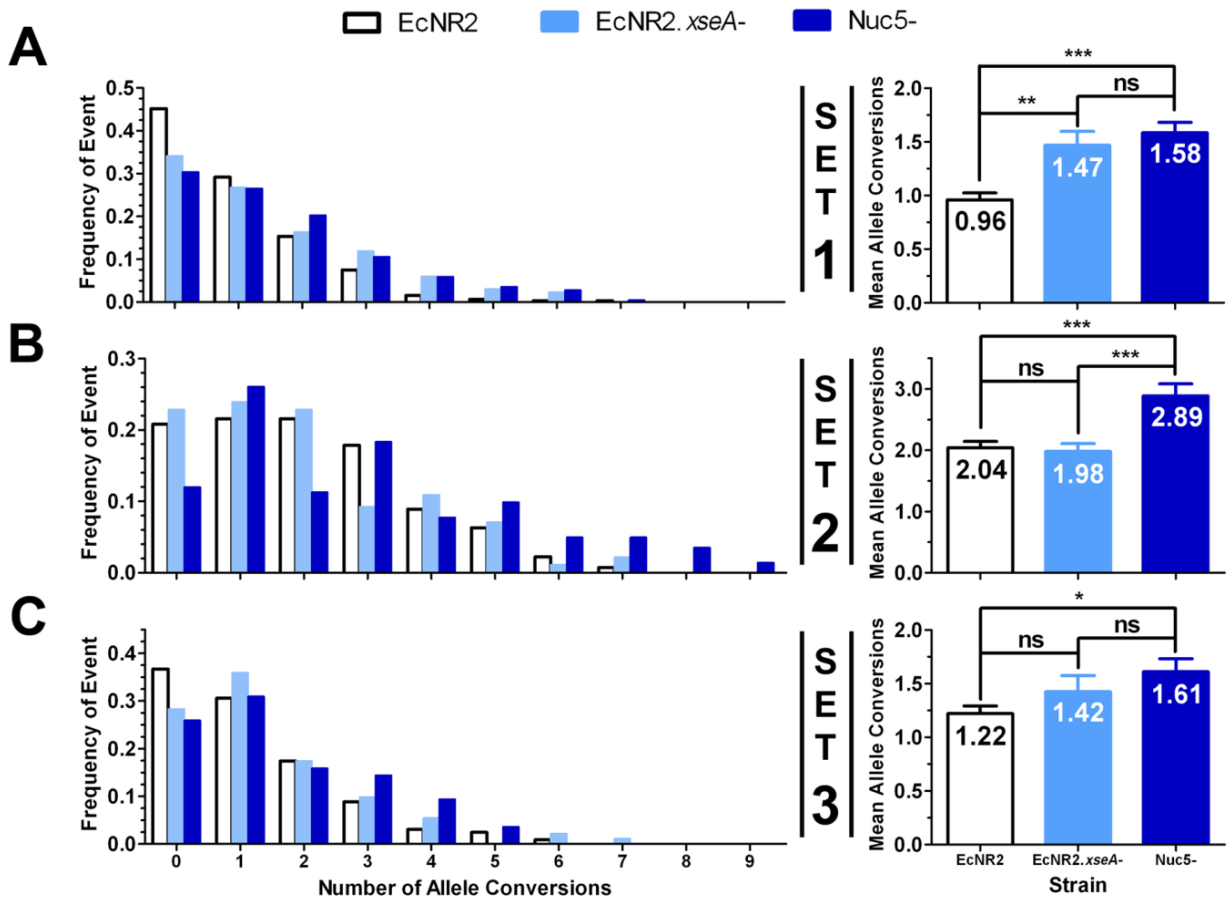


Figure 3-3. Effect of Nuclease Genotype on CoS-MAGE Performance. CoS-MAGE was carried out in three strains (EcNR2, EcNR2.*xseA*$^-$, and Nuc5$^-$), using sets of ten oligos encoding UAG→UAA mutations, and a co-selection oligo designed to revert a mutated selectable marker located within 500 kb of the targeted loci. **(A)** Set 1 was co-selected with chloramphenicol acetyltransferase (*cat*, inserted at the *mutS* locus). In comparison with EcNR2 (n = 319), both EcNR2.*xseA*$^-$ (** P = 0.0001, n = 135) and Nuc5$^-$ (*** P < 0.0001, n = 257) show statistically significant increases in mean oligo conversion, a decreased proportion of clones exhibiting no allele conversions, and more clones with 5+ conversions. **(B)** Set 2 was co-selected with beta lactamase (*bla*, inserted with the λ prophage). Here, Nuc5$^-$ (n = 142) shows a statistically significant increase in recombineering performance compared to both EcNR2 (*** P < 0.0001, n = 268) and EcNR2.*xseA*$^-$ (*** P < 0.0001, n = 184). **(C)** Set 3 was co-selected with *tolC*. Here, Nuc5$^-$ (n = 139) shows a statistically significant increase in mean allele conversion compared to EcNR2 (* P = 0.002, n = 327). EcNR2.*xseA*$^-$ (n = 92) shows an intermediate phenotype between EcNR2 (P = 0.2) and Nuc5- (P = 0.3). All oligos used in this experiment had two PT bonds on both ends. Data shown in the right panels are presented as the mean with the standard error of the mean. Statistical significance is denoted using a starred system where ns denotes a non-significant variation, * denotes P < 0.003, ** denotes P < 0.001, and *** denotes P < 0.0001.

0.2, respectively). Given that Set 1 exhibited the largest difference in performance between EcNR2 and Nuc5$^-$ (65% higher allele conversion in Nuc5$^-$), it is possible that Set 1 is the most susceptible to nuclease repression, and therefore that the effect of removing ExoVII would be most apparent for this set. Overall, using these three tested sets, Nuc5$^-$ is superior to

EcNR2.*xseA*$^{-}$. This suggests that the action of ExoVII somewhat compromises CoS-MAGE frequency, but that some or all of the other exonucleases removed in Nuc5^{-} also have a role in oligo degradation.

Impaired primase activity enhances multiplex allele replacement frequency: It is generally accepted that Redβ mediates annealing of exogenous DNA to the lagging strand of the replication fork prior to extension as a nascent Okazaki Fragment (*6, 12, 20, 21*). Therefore, we sought to increase the amount of ssDNA on the lagging strand by disrupting the ability of DnaG primase to initiate OFs. Prior work (*18*) has shown that DnaG K580A and Q576A mutations increase OF length *in vitro* by approximately 1.5-fold and 8-fold, respectively (see Table S3-2 for further explanation).

To investigate whether longer OFs could improve CoS-MAGE, we compared the performance of EcNR2, EcNR2.dnaG.K580A, EcNR2.dnaG.Q576A, and Nuc5-.dnaG.Q576A using Sets 1-3 as described above. EcNR2.dnaG.Q576A robustly outperformed EcNR2, yielding a significantly increased mean number of alleles converted (mean ± std. error of mean) for Set 1 (Figure 3-4, left panel, 1.43 ± 0.12 vs. 0.96 ± 0.07, ** P = 0.0003), Set 2 (Figure 3-4, middle panel, 2.63 ± 0.13 vs. 2.04 ± 0.10, ** P = 0.0003), and Set 3 (Figure 3-4, right panel, 2.54 ± 0.14 vs. 1.22 ± 0.07, *** P < 0.0001). Additionally, EcNR2.dnaG.K580A (intermediate-sized OFs) appears to have intermediate performance between EcNR2 (normal OFs) and EcNR2.dnaG.Q576A (longest OFs). This suggests that OF length correlates with AR frequency, and supports our hypothesis that exposing more ssDNA at the lagging strand of the replication fork enhances λ Redβ-mediated annealing.

Visualizing AR frequency for individual alleles in all three Sets (Figure 3-4C) reinforces the relationship between OF size and MAGE performance. Compared to EcNR2, the K580A

variant trends toward a modest increase in individual AR frequency, whereas the Q576A variant starkly improves AR frequency. Finally, the Nuc5-.dnaG.Q576A strain yielded the highest observed AR frequencies for all oligo sets, suggesting a combined effect of decreasing oligo

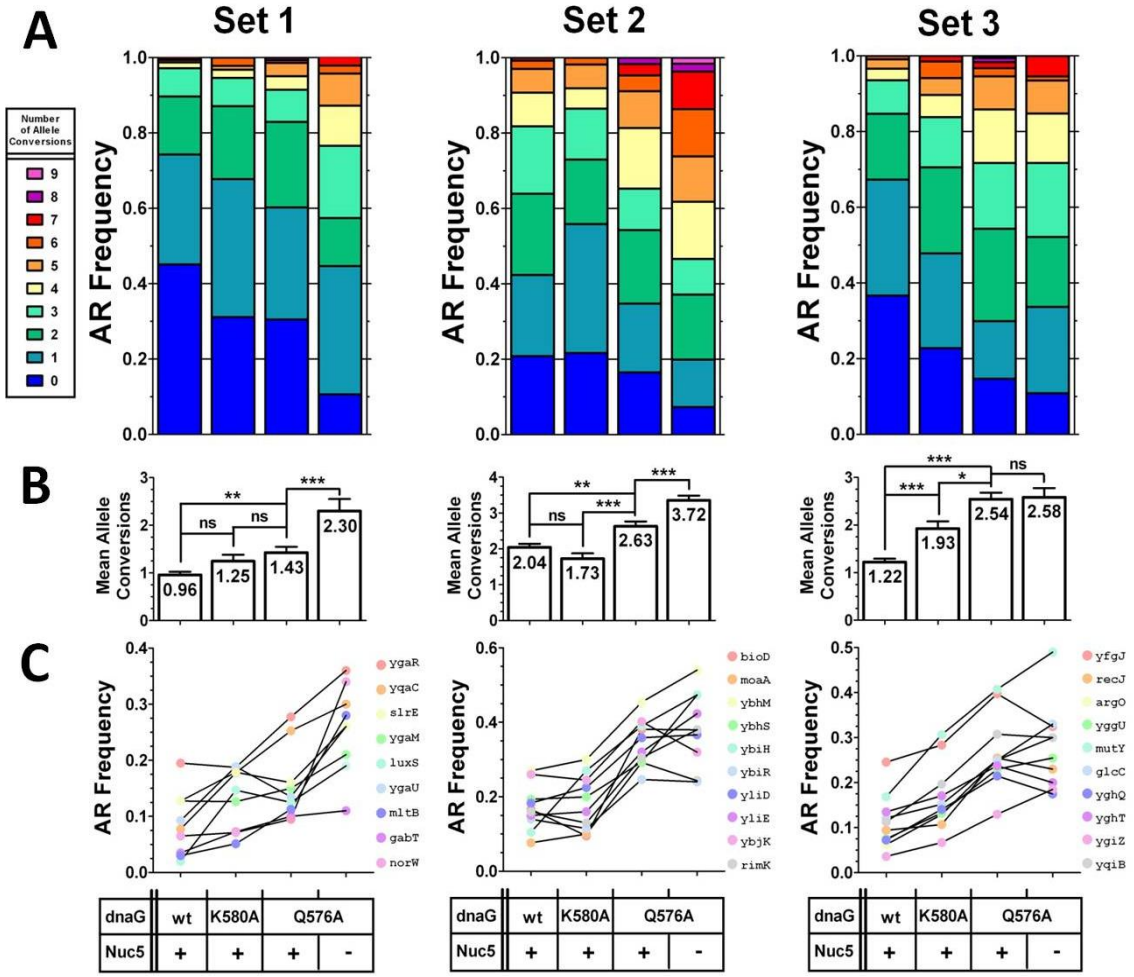


Figure 3-4. DnaG variants improve CoS-MAGE Performance. EcNR2, EcNR2.dnaG.K580A, EcNR2.dnaG.Q576A, and Nuc5-.dnaG.Q576A were tested for their performance in CoS-MAGE using three sets of 10 oligos as described in Figure 3-2B. For each set, all 10 alleles were simultaneously assayed by mascPCR in recombinant clones after one cycle of CoS-MAGE. (**A**) The data are presented using stacked AR frequency plots, which show the distribution of clones exhibiting a given number of allele conversions. (**B**) Mean number of alleles converted for each strain are shown with P-values indicated above the bars. Statistical significance is denoted using a star system where * denotes $P < 0.003$, ** denotes $P < 0.001$, and *** denotes $P < 0.0001$. The data are presented as the mean (reported numerically inside each bar) ± standard error of the mean. (**C**) AR frequencies for each individual allele are shown for all tested strains. Overall, the relative performance of each strain was Nuc5-.dnaG.Q576A > EcNR2.dnaG.Q576A > EcNR2.dnaG.K580A > EcNR2. This trend reflects an improvement commensurate with the severity of primase attenuation (*i.e.*, the Q576A variant has more severely disrupted primase and larger OFs than the K580A variant). Furthermore, Nuc5-.dnaG.Q576A combines the benefits of the DnaG Q576A variant and the benefits of the inactivation of 5 potent exonucleases (Mosberg, J.A., Gregg, C.J., *et al.*, in review). For Set 1: EcNR2, n=319; EcNR2.dnaG.K580A, n=93; EcNR2.dnaG.Q576A, n=141; Nuc5$^-$.dnaG.Q576A, n=47. For Set 2: EcNR2, n=269; EcNR2.dnaG.K580A, n=111; EcNR2.dnaG.Q576A, n=236; Nuc5$^-$.dnaG.Q576A, n=191. For set 3: EcNR2, n=327; EcNR2.dnaG.K580A, n=136; EcNR2.dnaG.Q576A, n=184; Nuc5$^-$.dnaG.Q576A, n=92.

degradation through nuclease inactivation and increasing the amount of exposed target ssDNA at the lagging strand of the replication fork. Interestingly, EcNR2.dnaG.Q576A strongly outperformed Nuc5- for Set 3 (*** $P < 0.0001$), whereas EcNR2.dnaG.Q576A performance was not significantly different than that of Nuc5- for Sets 1 ($P = 0.33$) and 2 ($P = 0.26$) (Tables 3-1 and 3-2). This suggests that the relative importance of replication fork availability and oligo protection can vary for MAGE targets throughout the genome, possibly due to oligo and/or locus-specific effects that have not yet been elucidated. Since both factors are important, combining impaired primase mutants with nuclease knockouts should reliably improve CoS-MAGE performance.

Table 3-1. Summary of mean number of alleles converted per clone for each MAGE oligo set

Set	EcNR2	Nuc5-	EcNR2.dnaG.Q576A	Nuc5-.dnaG.Q576A
	Mean ± SEM (n)	Mean ± SEM (n)	Mean ± SEM (n)	Mean ± SEM (n)
1	0.96 ± 0.07 (319)	**1.58 ± 0.10 (257)**	1.43 ± 0.12 (141)	2.30 ± 0.25 (92)
2	2.04 ± 0.10 (269)	**2.89 ± 0.19 (142)**	2.63 ± 0.13 (236)	3.72 ± 0.17 (191)
3	1.22 ± 0.07 (327)	1.61 ± 0.12 (139)	**2.54 ± 0.14 (184)**	2.59 ± 0.19 (92)

Table 3-2. CoS-MAGE Allele Replacement performance of modified strains (presented as fold change from EcNR2)

Metric	Set	Nuc5-	E2.dnaG.Q576A	Nuc5-.dnaG.Q576A
Average	1	1.65	1.49	2.40
	2	1.41	1.29	1.82
	3	1.32	2.08	2.12
	Average	**1.46**	**1.62**	**2.11**
5+ Conversions	1	5.28	3.96	10.18
	2	2.65	2.01	4.11
	3	1.07	4.20	4.52
	Average	**3.00**	**3.39**	**6.27**
0 Conversions	1	0.67	0.68	0.24
	2	0.58	0.79	0.35
	3	0.71	0.40	0.30
	Average	**0.65**	**0.62**	**0.29**

Okazaki fragment location is not a major determinant of available ssDNA on the lagging strand of the replication fork: Given the significant enhancement of CoS-MAGE performance in EcNR2.dnaG.Q576A, we sought to determine whether localizing all 10 targeted alleles to a single putative OF would result in "jackpot" recombinants with all 10 alleles converted. We hypothesized that nascent Okazaki Fragments sometimes obstructed target alleles, leading to a non-accessible lagging strand. According to this hypothesis, successful replacement of one allele would indicate permissive OF localization, greatly increasing the chance that other alleles occurring within the same OF could be replaced. Therefore, we speculated that the larger OF size in EcNR2.dnaG.Q576A might allow many changes to occur within 1 large OF. To test this, we designed 10 MAGE oligos that introduce inactivating nonsense mutations into a region spanning 1829 bp of *lacZ*. Despite their close proximity, all 10 alleles were spaced far enough

apart that their corresponding MAGE oligos would not overlap. Given the difference in average OF sizes between strains, it is unlikely for all 10 alleles to be located in the same OF in EcNR2, but quite likely that all 10 alleles would be located in the same OF in EcNR2.dnaG.Q576A. A *tolC* cassette (T.co-lacZ) was installed ~50 kb upstream of *lacZ* for efficient co-selection. Prior to use, this cassette was inactivated using the tolC-r_null_mut* oligo. Since the placement of these mutations is not compatible with mascPCR analysis, we used Sanger sequencing for analysis of white colonies. Blue colonies were scored as having zero conferred mutations. For EcNR2, 59% of the clones were white with 1.24 ± 0.23 (mean ± standard error of the mean) conversions per clone, whereas 84% of the EcNR2.dnaG.Q576A clones were white with 2.52 ± 0.25 allele conversions per clone (Figure 3-5). While EcNR2.dnaG.Q576A exhibits more mean allele conversions in CoS-MAGE than EcNR2 (*** $P < 0.0001$), the magnitude of this improvement (Figure 3-5B) is comparable with those observed for Sets 1-3 (Figure 3-4) where non-selectable oligos were spread across 70, 85, and 162 kb, respectively. Moreover, "jackpot" clones with 7+ converted alleles were not frequently observed for EcNR2.dnaG.Q576A using this oligo set. Thus although replication fork position is relevant, OF placement is not the predominant limiting factor for multiplex allele replacement. Other important factors could include target site occlusion by single stranded binding proteins or the availability of oligos, Redβ recombinase, or host factors.

Improved strains have larger optimal oligo pool size for multiplex allele replacement: A MAGE oligo pool size of approximately 10 was found to be most effective in prior studies (*7*). However, given the enhanced Redβ-mediated recombination in our Nuc5- and *dnaG* strains, we tested whether an expanded set of oligos would lead to more alleles converted in average and top clones. Therefore, we designed 10 additional MAGE oligos (Set 3X) that

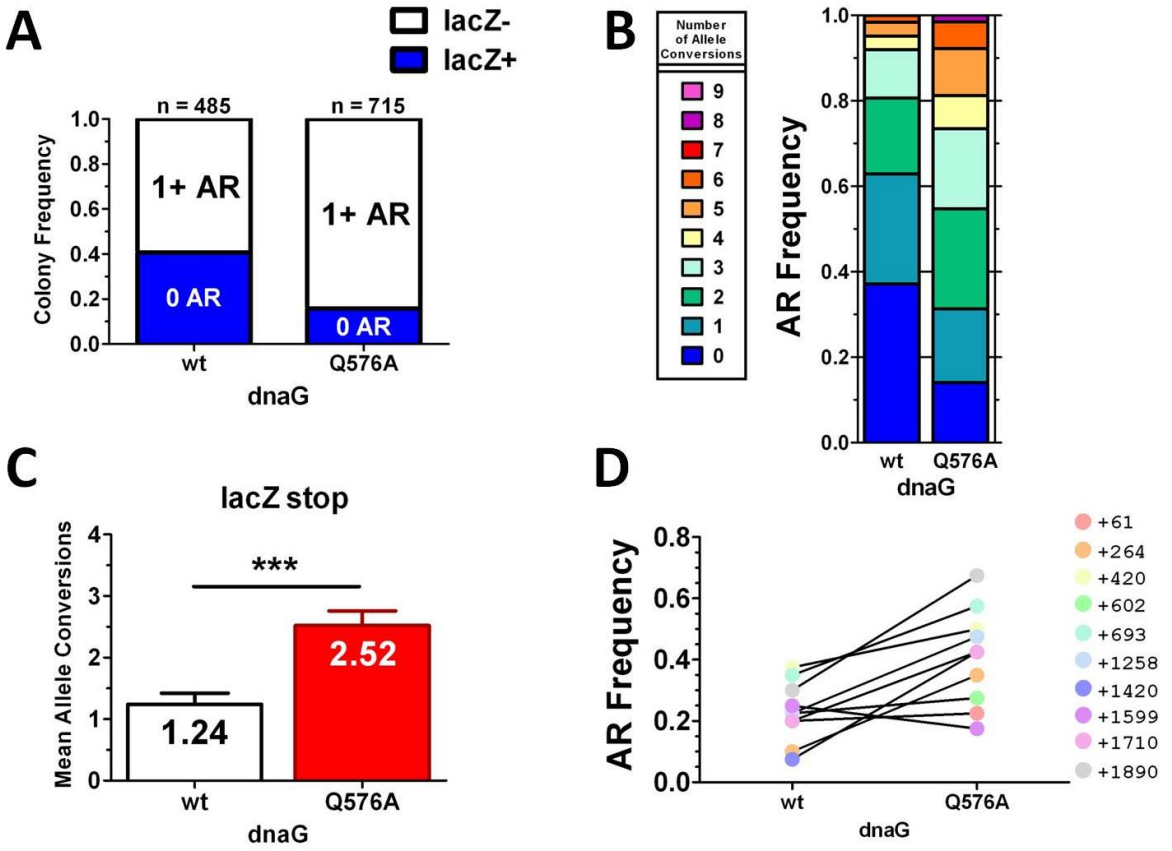


Figure 3-5. Placing all targeted alleles within one Okazaki Fragment does not cause a bimodal distribution for recombination frequency. EcNR2 and EcNR2.dnaG.Q576A were tested for their performance in CoS-MAGE using a set of 10 non-overlapping oligos that introduce 10 premature stop codons in the first 1,890 bp of *lacZ*. The targeted region of the genome is likely to be small enough to be frequently encompassed within a single Okazaki Fragment in EcNR2.dnaG.Q576A. After one cycle of CoS-MAGE, LacZ$^-$ recombinant clones were Sanger sequenced to assay all 10 alleles. Recombinations were performed in triplicate to estimate the frequency of white colonies (*lacZ*$^-$), but sequencing was only performed on a single replicate. (**A**) EcNR2.dnaG.Q576A (n=715, 5.33:1) exhibited a significant increase in the *lacZ*$^-$:*lacZ*$^+$ ratio compared to EcNR2 (n=485, 1.46:1). (**B**) EcNR2.dnaG.Q576A exhibited an AR distribution similar to those observed with Sets 1-3 (which span 70 kb, 85 kb, and 162 kb, respectively). (**C**) Compared to EcNR2, EcNR2.dnaG.Q576A exhibited a higher mean number of alleles converted (unpaired t-test, *** $P < 0.0001$). For EcNR2, n = 39, and for EcNR2.dnaG.Q576A, n = 55. (**D**) Compared to EcNR2, AR frequencies increased for 9 out of 10 individual alleles in EcNR2.dnaG.Q576A. The alleles are represented by their positions in *lacZ* (*e.g.*, "+61" means that this oligo introduces a nonsense mutation by generating a mismatch at the 61st nucleotide of *lacZ*). Taken together, all of these results demonstrate improved CoS-MAGE in EcNR2.dnaG.Q576A compared to EcNR2, but no significant enhancement was obtained from targeting all oligos to a single putative OF.

swapped synonymous AGA and AGG codons in alleles within the same region targeted by the Set 3 oligos. The *ygfT* allele (Set 3X) was not successfully assayed by mascPCR, so a maximum of 19 allele replacements could be detected out of the 20 conversions attempted. One round of CoS-MAGE using the combined oligo Sets 3 and 3X with *tolC* as a selectable marker improved

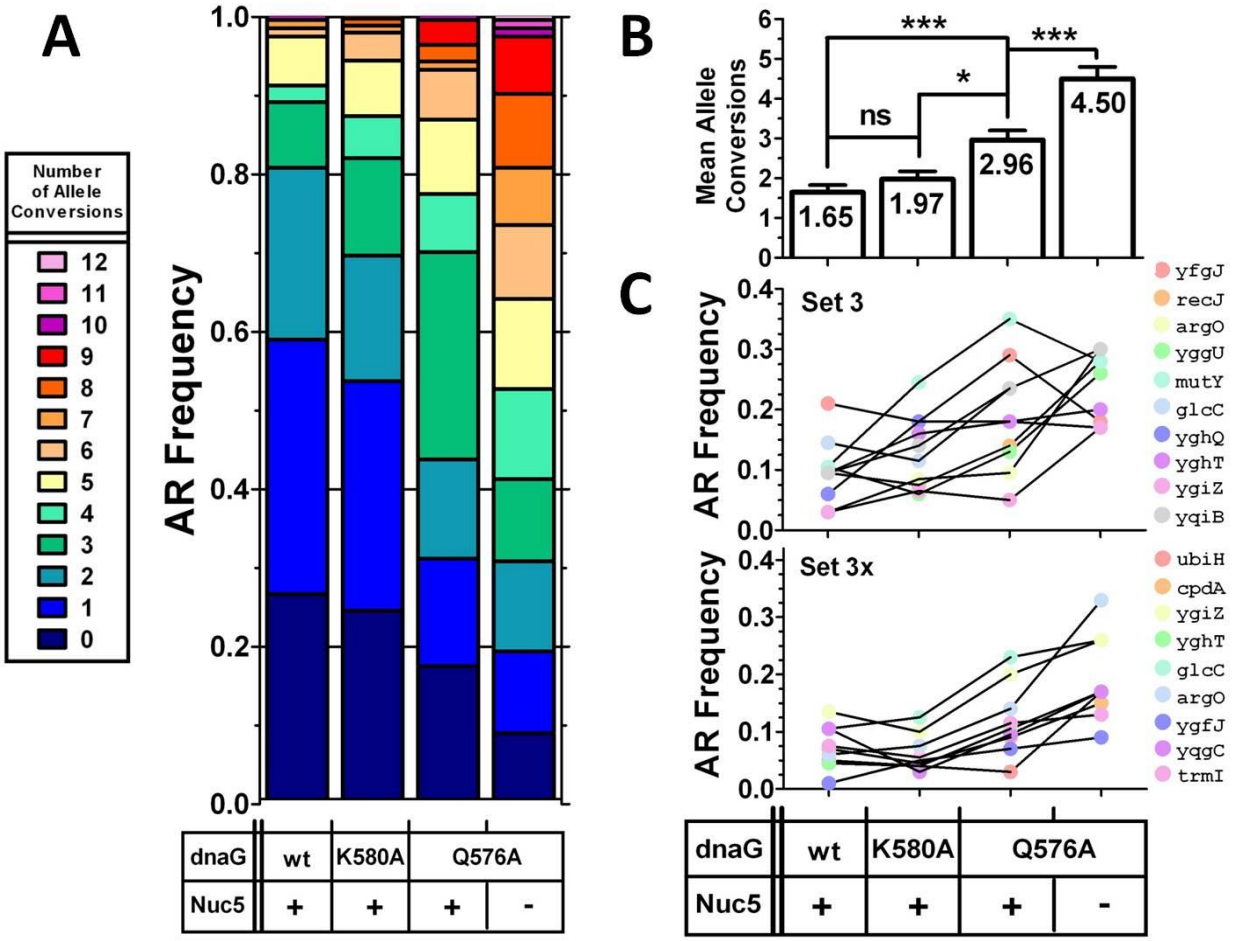


Figure 3-6. Testing DnaG variants with a 20-plex CoS-MAGE oligo set. EcNR2, EcNR2.dnaG.K580A, EcNR2.dnaG.Q576A, and Nuc5-.dnaG.Q576A were tested for their performance in CoS-MAGE using an expanded set of 20 oligos (Sets 3+3X). Genotypes of recombinant clones were assayed by mascPCR after one cycle of CoS-MAGE (*ygfT* could not be assayed by mascPCR). (**A**) AR frequency distributions. (**B**) Mean number of alleles converted ± standard error of the mean, with P-values indicated above the bars. Statistical significance is denoted using a star system where * denotes $P < 0.003$, ** denotes $P < 0.001$, and *** denotes $P < 0.0001$. (**C**) Mean individual AR frequencies. As seen with the smaller oligo sets, the *dnaG* variants reduce the number of clones with zero conversions and increase the average number of conversions per clone. Nuc5$^-$.dnaG.Q576A strongly outperforms all other strains, with a mean of 4.50 alleles converted and fewer than 10% of clones having zero conversions. Notably, Nuc5-.dnaG.Q576A has strongly improved performance with Sets 3+3X compared to Set 3, whereas EcNR2.dnaG.Q576A does not. EcNR2, n=96; EcNR2.dnaG.K580A, n=113; EcNR2.dnaG.Q576A, n=95; Nuc5$^-$.dnaG.Q576A, n=96.

AR frequency in all strains (Figure 3-6). The mean number of alleles converted (and fold increase over 10-plex means for Set 3 alone) per clone are as follows: 1.65 (1.35-fold) for EcNR2; 1.97 (1.02-fold) for EcNR2.dnaG.K580A; 2.96 (1.17-fold) for EcNR2.dnaG.Q576A; and 4.50 (1.74-fold) for Nuc5-.dnaG.Q576A (Figure 3-6B). Notably, Nuc5-.dnaG.Q576A exhibited the greatest improvement with the expanded oligo set, suggesting that preventing oligo

degradation is important when the intracellular concentration of each individual oligo is low. Longer OFs then increase the probability that scarce oligos will find their genomic target. This observation assumes that a limited number of oligos are internalized during electroporation, which is consistent with the fact that the mole fraction of an oligo in a multiplex experiment affects its relative AR frequency at saturating oligo concentrations (*3*). Notably, the Set 3X oligos yielded lower recombination frequencies compared to the Set 3 alleles that converted UAG codons to UAA, and Nuc5-.dnaG.Q576A strongly elevated the AR frequency of these alleles (Figure 3-6C). Nuc5-.dnaG.Q576A exhibited the largest number of simultaneous allele conversions reported to date in a single recombination (*tolC* plus 12 additional alleles converted). Although CoS-MAGE in Nuc5-.dnaG.Q576A was able to simultaneously convert an unprecedented number of alleles, the lack of clones with allele replacements near the maximum of 19 suggests that CoS-MAGE is approaching a practical maximum for oligo pool complexity, where further increases in oligo pool size may not substantially improve AR frequency or increase the mean number of alleles converted.

Disrupting DnaG primase activity enhances leading strand recombination: Since DnaG primase synthesizes RNA primers only at the lagging strand of the replication fork, we expected its alteration to have minimal effect on Redβ-mediated annealing to the leading strand. To examine this hypothesis, we tested oligos designed to target the Set 3 alleles on the leading strand (reverse complements of the Set 3 oligos described above). The *tolC*-reverting co-selection oligo was also re-designed to target the leading strand so that the correct strand would be co-selected. Although the number of *tolC*-reverted co-selected recombinants were few, of the *tolC*+ clones, EcNR2 gave 0.85 ± 0.13 allele conversions per clone (mean ± std. error of the mean, n = 88), whereas EcNR2.dnaG.Q576A gave 1.39 ± 0.18 conversions (n = 91), which was

significantly different (P = 0.018). Similar to lagging targeting Set 3, we observed a reduction in zero conversion events for EcNR2.dnaG.Q576A, as well as a broadening of the distribution of total allele conversions per clone and a greater maximum number of alleles converted (Figure S3-1A). Thus, leading-targeting CoS-MAGE yields recombination frequencies nearly within two-fold of those attained with lagging-targeting CoS-MAGE (1.22 ± 0.07 vs. 2.54 ± 0.14 for EcNR2 and EcNR2.dnaG.Q576A, respectively). Furthermore, contrary to our expectations, EcNR2.dnaG.Q576A exhibited significantly enhanced AR frequency over EcNR2 at 9 out of 10 alleles on the leading strand (Figure S3-1C). Interestingly, using leading targeting oligos, the co-selection advantage quickly diminished with distance (Figure S3-1B, top panel). In contrast, co-selection using lagging targeting oligos increases the AR frequency of other alleles spanning a large genomic distance (~0.5 Mb; (*3*)), as observed for the lagging-targeting Set 3 oligos (Figure S3-1B, bottom panel).

Disrupting DnaG primase activity enhances deletions but not insertions: MAGE is most effective at introducing short mismatches, insertions, and deletions, as these can be efficiently generated using λ Red-mediated recombination without direct selection (*1*). However, large deletions and gene-sized insertions are also important classes of mutations that could increase the scope of applications for MAGE. For example, combinatorial deletions could be harnessed for minimizing genomes (*22*) and efficient insertions could increase the ease of building biosynthetic pathways by removing the need for linking inserted genes to selectable markers (*23-26*). Large deletions require two separate annealing events often spanning multiple OFs, but large insertions should anneal within the same OF, as the heterologous portion loops out and allows the flanking homologies to anneal to their adjacent targets (*12, 20*). Maresca et al. (*20*) have demonstrated that the length of deletions have little effect on Redβ-mediated

recombination, but that insertion frequency is highly dependent on insert size (presumably due to constraints on λExo-mediated degradation of the leading-targeting strand and not the lagging-targeting strand). Therefore, we investigated whether diminishing DnaG primase function would enhance deletion and/or insertion frequencies.

Based on the ssDNA intermediate model for λRed recombination (*12*, *20*), we expected enhanced deletion frequency in EcNR2.dnaG.Q576A especially for intermediate-sized deletions (500 bp – 10 kb), since less frequent priming would increase the probability of both homology regions being located in the same OF. Therefore, we designed three oligos that deleted 100 bp, 1,149 bp, or 7,895 bp of the genome, including a portion of *galK*. In addition to *galK*, oligo galK_KO1.7895 deleted several nonessential genes (*galM*, *gpmA*, *aroG*, *ybgS*, *zitB*, *pnuC*, and *nadA*). The recombined populations were screened for the GalK$^-$ phenotype (white colonies) on MacConkey agar plates supplemented with galactose as a carbon source. EcNR2.dnaG.Q576A significantly outperformed EcNR2 for the 100 bp (* P = 0.03) and 1,149 bp (* P = 0.03) deletions, but there was no difference detected between the two strains for the 7,895 bp deletion (P = 0.74, Figure S3-2). The lack of improvement using galK_KO1.7895 may be due to reduced target availability if the two homology sites are split across two or more OFs even in EcNR2.dnaG.Q576A.

Finally, if λExo degradation most strongly impacts λRed-mediated insertions of large cassettes, modifying the replisome should not significantly impact their insertion frequency. Therefore, we quantified the insertion frequency of a selectable kanamycin resistance cassette (*lacZ*::*kanR*, 1.3 kb) targeted to *lacZ*. Insertion of *lacZ::kanR* (*1*, *12*) in three replicates yielded recombination frequencies of 1.81E-04 ± 6.24E-05 in EcNR2 versus 1.28E-04 ± 4.52E-05 in

EcNR2.dnaG.Q576A (P = 0.30 by unpaired t-test). Therefore, modifying DnaG primase function does not appear to significantly affect λ Red-mediated gene insertion.

Discussion

MAGE is a powerful technique that can be used to generate combinatorial sets of designed mutations in a population (*1*) and/or modify hundreds of alleles in a single strain (*7*). We have engineered optimized strains for multiplex genome engineering in an effort to streamline extensive genome editing. Previously, we showed that converting a selectable allele in the vicinity of multiple non-selectable alleles enriches the candidate pool for highly modified clones (*3*). In this work, we demonstrated that exonucleases are capable of degrading single stranded MAGE oligos even when these oligos are protected using phosphorothioate bonds. Inactivating ExoI, ExoVII, ExoX, RecJ, and λExo significantly enhanced multiplex AR frequencies. This showed that intracellular MAGE oligos are a limiting factor in λ Redβ-mediated recombination. We also demonstrated that available ssDNA on the lagging strand of the replication fork is a limiting factor that can be increased by disrupting the interaction between DnaG primase and DnaB helicase on the replisome.

In order to increase ssDNA on the lagging strand of the replication fork, we introduced two known mutations in primase (DnaG)—K580A and Q576A. These mutations have been shown *in vitro* to increase OF size by interrupting the primase-helicase interaction on the replisome (*18*). Based on the measurements of Tougu *et al.* (*18*), we estimate that the K580A mutation increases OF length by ~1.5-fold and the Q576A mutation increases OF length by ~8-fold (see Table S3-2). EcNR2.dnaG.K580A and EcNR2.dnaG.Q576A exhibited significant increases in the mean number of alleles converted and decreases in the proportion of clones with

zero non-selectable alleles converted. Furthermore, the strongest enhancement was observed in EcNR2.dnaG.Q576A (the variant with the longest OFs of the strains reported herein), with an intermediate enhancement observed in EcNR2.dnaG.K580A (the variant with intermediate-sized OFs). This relationship between recombination frequency and OF length further supports the model in which Redβ mediates annealing at the lagging strand of the replication fork (*6, 12, 20, 21*), and our hypothesis that ssDNA on the lagging strand of the replication fork is a limiting factor during this process. With this in mind, we unsuccessfully attempted to generate a DnaG Q576A/K580A double mutant, suggesting that such an extensive manipulation of the DnaG C-terminal helicase interaction domain (*27*) was lethal.

Our results indicate that intracellular concentrations of MAGE oligos and the accessibility of their genomic targets are both limiting. To further increase the number of simultaneous mutations that can be generated by CoS-MAGE, it is helpful to understand whether the AR frequency is limited predominantly by the number of oligos that enter the cytoplasm, or whether kinetics are also relevant. Since a maximum of 9 allele replacements was observed for the 10-oligo sets compared to a maximum of just 12 allele replacements for the 20-oligo set, oligo uptake may be limiting. However, the fact that primase modulation—in addition to nuclease inactivation—enhances AR frequency underscores the kinetic constraints regarding Redβ-mediated annealing. Each missed opportunity to anneal 1) increases the number of wt alleles in the population due to replication and 2) decreases the number of MAGE oligos available, via dilution (cell division) and degradation (nucleases). Increasing the concentration of each reactant (*i.e.*, intracellular oligos and accessible genomic targets) would increase the kinetics of annealing. Therefore, the number of intracellular oligos may limit the maximum

number of possible mutations, but kinetics appear to be a significant force limiting the population-wide AR frequency average.

Interestingly, the nuclease-deficient Nuc5- strain (Mosberg, J.A., Gregg, C.J., *et al.*, in review) performed statistically similarly to the EcNR2.dnaG.Q576A strain for Sets 1 and 2, while EcNR2.dnaG.Q576A strongly outperformed the nuclease-deficient strain for Set 3 (Tables 3-1 and 3-2). While oligo design parameters such as type of designed mutation (*1*), oligo length (*1*), oligo secondary structure (*1*) and off-target genomic homology (*7*) are major determinants of AR frequency, our results highlight the relevance of genomic context. This has previously been difficult to demonstrate, but is apparent from the discrepancy in performance of the same oligo sets tested in Nuc5-, EcNR2.dnaG.Q576A, and Nuc5-.dnaG.Q576A. For example, different regions may have different replication fork speed or priming efficiency. These factors could locally modulate OF length, thus affecting Redβ-mediated AR frequency (although replication fork speed did not appear to be a major factor *in vitro* (*18*)). Therefore, increasing the region that must be replicated by a single OF may profoundly increase AR frequency for oligos targeting such regions. Alternatively, certain oligos may be more susceptible to nuclease degradation, so removing the responsible nucleases would disproportionately improve AR frequency for such oligos. With this in mind, we tested whether combining primase modification and nuclease removal would enhance MAGE performance more than either strategy used individually. Indeed, Nuc5-.dnaG.Q576A consistently performed the best (Figures 3-4 and 3-6) of all tested strains. Therefore, these two disparate strategies can be combined for a larger and more robust MAGE enhancement (Figure 3-7).

To explore the extent to which OF localization impacts CoS-MAGE performance, we tested whether placing 10 oligos within a single putative OF would yield subpopulations of

unmodified (few alleles converted) and "jackpot" (most alleles converted) recombinants. However, CoS-MAGE using the densely-clustered *lacZ* oligos (Figure 3-5) produced a similar AR distribution to the ones observed for Sets 1-3 (Figure 3-4), which target regions of the genome spanning several putative OFs. Since mutations within a single putative OF behaved similarly to mutations spread across many OFs, nascent OF placement does not appear to be a critical determinant of multiplex AR frequency. A number of hypotheses could explain why the expected "jackpots" are not observed. Most likely, MAGE oligos are limiting due to degradation and/or lack of uptake. Thus, it is possible that most cells lack some of the oligos necessary for generating a majority of the desired mutations. Additionally, OF extension may occur too fast for all of the MAGE oligos to anneal before the OF occludes their targets. Still another explanation could be that ssDNA binding proteins occlude ssDNA on portions of the lagging strand, rendering these regions non-accessible for Redβ-mediated annealing. Finally, it is also possible that several MAGE oligos annealed within a single OF could destabilize lagging strand synthesis, leading to selection

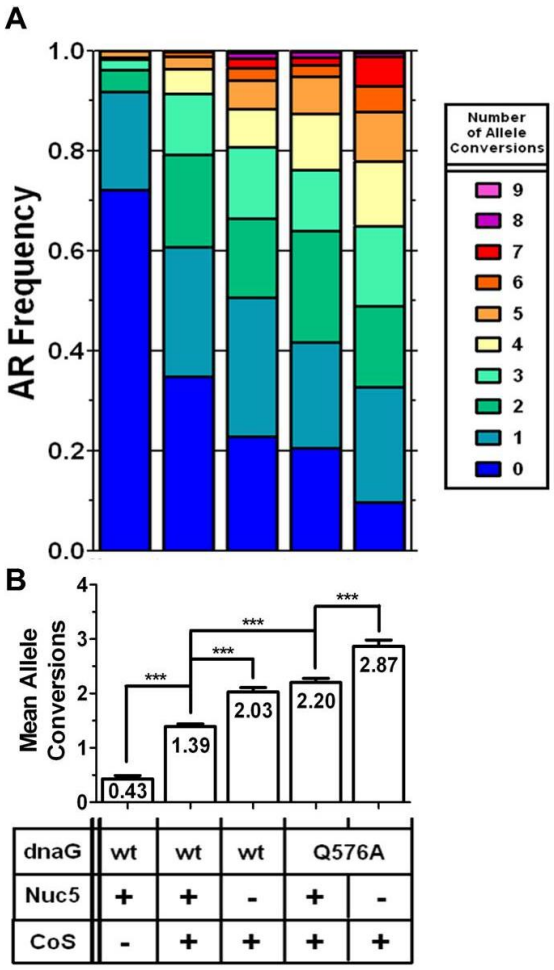


Figure 3-7. Averaged CoS-MAGE performance by strain. CoS-MAGE, nuclease deletion, and replisome manipulation all independently improve recombination frequency. **(A)** The distributions of the number of alleles converted per MAGE or CoS-MAGE cycle were averaged across Sets 1, 2, and 3 for each specified condition. These averaged results give the approximate expected performance of a 10-plex recombination. **(B)** The mean allele conversions per recombination is reported for each distribution that was shown in **(A)**.

against highly-modified "jackpot" clones. Indeed, Corn *et al.* (*28*) hypothesize that DnaG primase has evolved to only initiate synthesis when multiple DnaG units are bound to DnaB Helicase, as OF synthesis away from the replisome could be detrimental. Since $polIII_{lag}$ dissociates from the replisome after completing an OF (*29*), the rapid and repeated dissociation of $polIII_{lag}$ caused by multiple nearby MAGE oligos could inhibit lagging strand synthesis as the replisome proceeds beyond the target region. In the absence of the rest of the replisome, a cytosolic PolIII holoenzyme alone can synthesize 1.4 kb on a ssDNA template primed by 30 nt DNA oligos (*30*), but this activity is considerably diminished compared to that of an intact replisome. Therefore, if OFs are not completed while the replisome is in close proximity, this could result in persisting ssDNA that could destabilize the chromosome and/or cause lesions when the next replication fork passes through.

We also investigated whether targeting a greater number of alleles would increase the resulting number of conversions in our enhanced strains (Figure 3-6). Although the mean number of alleles converted (mean ± std. error of the mean) increased from 2.59 ± 0.19 with 10-oligo Set 3 to 4.50 ± 0.30 (1.74-fold) with 20-oligo Sets 3+3X for Nuc5-.dnaG.Q576A, the mean number of alleles converted for EcNR2.dnaG.Q576A only increased from 2.54 to 2.96 (1.17-fold). The superior enhancement for the nuclease-depleted Nuc5-.dnaG.Q576A strain suggests that the intracellular oligo concentration is a limiting factor for highly multiplexed MAGE (>10 alleles targeted). Therefore, enhancing DNA uptake and/or preservation may be a fruitful means of further improving MAGE. However, the greater multiplexibility of Nuc5-.dnaG.Q576A could also be due to the 10 new Set 3X oligos being more responsive to decreased exonuclease degradation than to increased lagging strand ssDNA availability. Additionally, there may be other limiting factors such as insufficient Redβ or unidentified host proteins. Although there is

no known precedent for limiting amounts of λ Red proteins during recombination (*31*), our new ability to attain 12 simultaneous non-selectable allele replacements (Figure 3-6A) shows that our improved strains are in uncharted territory for probing the limits of λ Red recombination.

Given that DnaG primase acts solely on the lagging strand of the replication fork, we expected that the primase modifications would only enhance lagging strand recombination. Therefore, the performance of leading-targeting CoS-MAGE in our strains was surprising, as EcNR2.dnaG.Q576A significantly outperformed EcNR2 (P = 0.018). Furthermore, while the total number of *tolC*+ recombinants was far smaller (~10^2-fold) for leading-targeting CoS-MAGE, the AR frequency of non-selectable alleles in these recombinants was still quite impressive, especially in extremely close proximity to the selectable allele. This suggests that one leading strand recombination event strongly correlates with multiple additional recombinations. Two possible explanations for the superior performance of EcNR2.dnaG.Q576A in leading-targeting CoS-MAGE are that 1) an impaired primase-helicase interaction increases accessible leading strand ssDNA, or 2) infrequent Redβ-mediated strand invasion initiates a new replication fork that travels in the opposite direction and swaps which strand is the lagging strand.

There is strong support for primase function affecting the dynamics of replication on both the lagging and leading strands (*29, 30, 32*). Lia *et al.* (*29*) observed phases in which OF synthesis is faster than helicase progression at the replication fork, alternating with phases in which helicase progression outstrips the rate of OF synthesis by $PolIII_{lag}$. These results demonstrate that DnaB-$PolIII_{lead}$ does not progress at the same instantaneous speed as $PolIII_{lag}$ (*29*). Furthermore, Yao *et al.* (*32*) showed that the velocity of leading-strand synthesis decreases during lagging strand synthesis, while its processivity increases. Perhaps less frequent primase-

helicase binding leads to transient asynchrony of the helicase and $PolIII_{lead}$. Given that PolIII tends to release from the replication fork more readily than does DnaB helicase (*32*), a transiently increased fork rate and decreased $PolIII_{lead}$ processivity could exacerbate such an asynchrony, creating a leading strand trombone loop similar to those observed during lagging strand synthesis. However, the effects of lagging strand synthesis on leading strand replication have been historically difficult to demonstrate in experiments beyond single-molecule studies (*32*). Given that instantaneous changes in replication dynamics appear to occur on timescales relevant to Redβ-bound oligo recombination, it is conceivable that snapshots of exposed ssDNA on the leading strand template could be recorded by measuring rates of leading-targeting AR. Single-molecule analysis of the Q576A variant could explore this hypothesis.

Alternatively, Redβ has been reported to facilitate strand invasion *in vitro* (*33*). If this also occurs *in vivo*, such strand invasion would produce a D-Loop that could act as a new origin of replication (*34*). Therefore, invasion of one leading-targeting MAGE oligo could initiate a replication fork traveling in the opposite direction. In the reverse orientation, the leading strand would become the lagging strand so that upstream oligos would become lagging targeting and much more likely to recombine. This could lead to the highly modified clones that we observed during leading-targeting CoS-MAGE (Figure S3-1). If this is the case, the non-selectable alleles would be upstream of the *tolC* selectable marker. Since co-selection is most effective downstream of the selectable marker (*3*), this may explain why co-selection enhancements decay rapidly with distance on the leading strand.

In this manuscript, we have identified intracellular MAGE oligo stability and availability of ssDNA on the lagging strand of the replication fork as a limiting factor in multiplex genome engineering. Compared to a single round of CoS-MAGE with ten synthetic oligonucleotides in a

standard recombineering strain (EcNR2), Nuc5- displays on average 46% more alleles converted per clone, 200% more clones with five or more allele conversions, and 35% fewer clones without any allele conversions. Additionally, EcNR2.dnaG.Q576A displays on average 62% more alleles converted per clone, 239% more clones with five or more allele conversions, and 38% fewer clones with zero allele conversions (Table 3-2). We combined these strain enhancements, generating Nuc5-.dnaG.Q576A, which has extended Okazaki Fragments and reduced exonuclease activity. These modifications exploited two distinct mechanisms that together increased the robustness and potency of CoS-MAGE, enabling an average of 4.50 and a maximum of 12 allele replacements in single cells exposed to a pool of 20 different synthetic allele replacement oligos (Figure 3-6). Additionally, 48% of recombinants had five or more allele replacements and only 8% had zero modified non-selectable alleles. Furthermore, in a given round of CoS-MAGE with ten synthetic oligos, Nuc5-.dnaG.Q576A displays on average 111% more alleles converted per clone, 527% more clones with five or more allele conversions, and 71% fewer clones with zero allele conversions in comparison with EcNR2 (Table 3-2). This improvement in MAGE performance will be highly valuable for increasing the diversity explored during the directed evolution of biosynthetic pathways (*1*) and for enabling the rapid generation of desired genotypes involving tens to hundreds of allele replacements (*7*).

Materials and Methods

Table S3-1 presents a full list of DNA oligos used in this work. All oligos were ordered with standard purification and desalting from Integrated DNA Technologies. Cultures were grown in LB-Lennox media (LB^L; 10 g tryptone, 5 g yeast extract, 5 g NaCl per 1 L water).

Strain Creation: Oligo-mediated λ Red recombination was used to generate all mutations as described below. All of the strains used in this work were generated from EcNR2 (*1*) {*E. coli* MG1655 Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*] Δ*mutS*::*cat*}. Strain Nuc5- was generated using knockout oligos (Table S3-1) that introduced a premature stop codon and a frameshift mutation at the beginning of the nuclease gene, thereby rendering the nuclease inactive. Strain Nuc5-.dnaG.Q576A was generated by recombining oligo dnaG_Q576A into strain Nuc5-. EcNR2.DT was created by deleting the endogenous *tolC* gene using the tolC.90.del recombineering oligo (*7*). EcNR2.T.co-lacZ was created by recombining a *tolC* cassette (T.co-lacZ) into the genome of EcNR2.DT, upstream of the lac operon. CoS-MAGE strains were prepared by inactivating a chromosomal selectable marker (*cat*, *tolC*, or *bla*) using a synthetic oligo. Clones with a sensitivity to the appropriate antibiotic or SDS (*25*) were identified by replica plating. The growth rate of strains EcNR2, EcNR2.dnaG.K580A, and EcNR2.dnaG.Q576A are approximately equivalent, while Nuc5-.dnaG.Q576A has a doubling time that is only ~7% longer than the others.

Generating dsDNA Recombineering Cassettes: The T.co-lacZ dsDNA recombineering cassette was generated by PCR using primers 313000.T.lacZ.coMAGE-f and 313001.T.lacZ.coMAGE-r (Table S3-1). The PCR was performed using KAPA HiFi HotStart ReadyMix, with primer concentrations of 0.5 μM and 1 μL of T.5.6 (*7*) used as template (a terminator was introduced downstream of the stop codon in the *tolC* cassette). PCRs (50 μL total) were heat activated at 95 °C for 5 min, then cycled 30 times at 98 °C (20 sec), 62 °C (15 sec), and 72 °C (45 sec). The final extension was at 72 °C for 5 min. The Qiagen PCR purification kit was used to isolate the PCR products (elution in 30 μL H_2O). Purified PCR products were quantitated on a NanoDrop™ ND1000 spectrophotometer and analyzed on a 1%

agarose gel with ethidium bromide staining to confirm that the expected band was present and pure.

***In vitro* dsDNA Digestion by Lambda Exo:** *LacZ::kanR* dsDNA (20 ng) with zero, one, or both ends phosphorothioated (VPT1, VPT2, and VPT4, respectively; see Figure S3-1A) was added to 9 µL of 1X Lambda Exonuclease Buffer (New England Biolabs). Lambda Exonuclease (New England Biolabs) was serially diluted in 1X Lambda Exonuclease Buffer as needed and 1 µL was then added to the reaction. Reactions were incubated at 37 °C for 30 min, heat inactivated at 75 °C for 10 min, and then analyzed on an Invitrogen 6% TBE non-denaturing PAGE gel (180 V, 40 min, post-stained in Invitrogen SYBR Gold for 15 min).

Performing λ Red Recombination: λ Red recombinations of ssDNA and dsDNA were performed as previously described (*1, 12*). Briefly, 30 µL from an overnight culture was inoculated into 3 mL of LB^L and grown at 30 °C in a rotator drum until an OD_{600} of 0.4-0.6 was reached (typically 2-2.5 hrs). The cultures were transferred to a shaking water bath (300 rpm at 42 °C) for 15 minutes to induce λ Red, then immediately cooled on ice for at least 3 minutes. For each recombination, 1 mL of culture was washed twice in ice cold deionized water (dH_2O). Cells were pelleted between each wash by centrifuging at 16,000 rcf for 20 seconds. The cell pellet was resuspended in 50 µL of dH_2O containing the DNA to be recombined. For recombination of dsDNA PCR products, 50 ng of PCR product was used. Recombination using dsDNA PCR products was not performed in Nuc5- strains, since λ Exo is necessary to process dsDNA into a recombinogenic ssDNA intermediate prior to β-mediated annealing (*12, 20*). For experiments in which a single oligo was recombined, 1 µM of oligo was used. For experiments in which sets of ten or twenty recombineering oligos were recombined along with a co-selection oligo, 0.5 µM of each recombineering oligo and 0.2 µM of the co-selection oligo were used (5.2 µM total for 10-

plex and 10.2 μM total for 20-plex). A BioRad GenePulser™ was used for electroporation (0.1 cm cuvette, 1.78 kV, 200 Ω, 25 μF), and electroporated cells were allowed to recover in 3 mL LB^L in a rotator drum at 30 °C for at least 3 hours before plating on selective media. For MAGE and CoS-MAGE experiments, cultures were recovered to apparent saturation (5 or more hours) to minimize polyclonal colonies (this was especially important for strains based on Nuc5-, which exhibit slow recovery after λ Red induction/electroporation). MAGE recovery cultures were diluted to ~5000 cfu/mL, and 50 μL of this dilution was plated on non-selective LB^L agar plates. To compensate for fewer recombinants surviving the co-selection, CoS-MAGE recovery cultures were diluted to ~1E5 cfu/mL and 50 μL of this dilution was plated on appropriate selective media for the co-selected resistance marker (LB^L with 50 μg/mL carbenicillin for *bla*, 20 μg/mL chloramphenicol for *cat*, or 0.005% w/v SDS for *tolC*). Leading-targeting CoS-MAGE recovery cultures were diluted to ~5E6 cfu/mL before plating.

Analyzing Recombination: *GalK* activity was assayed by plating recovered recombination cultures onto MacConkey agar supplemented with 1% galactose as a carbon source. Red colonies were scored as *galK*+ and white colonies were *galK*-. *LacZ* activity was assayed by plating recovery cultures onto LB^L agar + X-gal/IPTG (Fisher ChromoMax IPTG/X-Gal solution). Blue colonies were scored as *lacZ*+ and white colonies were *lacZ*-.

Kapa 2G Fast ReadyMix was used in colony PCRs to screen for correct insertion of dsDNA selectable markers. PCRs had a total volume of 20 μL, with 0.5 μM of each primer. These PCRs were carried out with an initial activation step at 95 °C for 2 min, then cycled 30 times at 95 °C (15 sec), 56 °C (15 sec), 72 °C (40 sec), followed by a final extension at 72 °C (90 sec).

Allele-specific colony PCR (ascPCR) was used to detect the nuclease and primase mutations. Multiplex allele-specific colony PCR (mascPCR) (*35*) was used to detect the 1-2 bp mutations generated in the MAGE and CoS-MAGE experiments. Each allele is interrogated by two separate PCRs—one with a forward primer whose 3' end anneals to the wild-type allele, and the other with a forward primer whose 3' end anneals to the mutated allele (the same reverse primer is used in both reactions). Primers are designed to have a T_m ~ 62 °C, but a gradient PCR is necessary to optimize annealing temperature (typically between 63 °C and 67 °C) to achieve maximal specificity and sensitivity for a given set of primers. A wild-type allele is indicated by amplification only in the wt-detecting PCR, while a mutant allele is indicated by amplification only in the mutant-detecting PCR. For mascPCR assays, primer sets for interrogating up to 10 alleles are combined in a single reaction. Each allele has a unique amplicon size (100 bp, 150 bp, 200 bp, 250 bp, 300 bp, 400 bp, 500 bp, 600 bp, 700 bp, and 850 bp). Template is prepared by growing monoclonal colonies to late-log phase in 150 uL LB^L and diluting 2 uL of culture into 100 uL dH2O. Typical mascPCR reactions use KAPA2GFast Multiplex PCR ReadyMix and 10X Kapa dye in a total volume of 10 µL, with 0.2 µM of each primer and 2 µL of template. These PCRs were carried out with an initial activation step at 95 °C (3 min), then cycled 27 times at 95 °C (15 sec), 63-67 °C (30 sec; annealing temperature optimized for each set of mascPCR primers), and 72 °C (70 sec), followed by a final extension at 72 °C (5 min). All mascPCR and ascPCR assays were analyzed on 1.5% agarose/EtBr gels (180 V, duration depends on distance between electrodes) to ensure adequate band resolution.

We performed at least two independent replicates for all strains with each oligo set in CoS-MAGE experiments. All replicates for a given strain and oligo set were combined to generate a complete data set. Polyclonal or ambiguous mascPCR results were discarded from our

analysis. Mean number of alleles replaced per clone were determined by scoring each allele as 1 for converted or 0 for unmodified. Given the sample sizes tested in the CoS-MAGE experiments ($n > 47$), we used parametric statistical analyses instead of their non-parametric equivalents, since the former are more robust with large sample sizes (*36*). We used a one way ANOVA to test for significant variance in CoS-MAGE performance of the strains (EcNR2, EcNR2.dnaG.K580A, EcNR2.dnaG.Q576A, Nuc5-, and Nuc5-.dnaG.Q576A) for a given oligo set. Subsequently, we used a Student's t-test to make pairwise comparisons with significance defined as $P < 0.05/n$, where n is the number of pairwise comparisons. Here, $n = 15$ as this data set was planned and collected as part of a larger set with 6 different strains although only EcNR2, EcNR2.dnaG.K580A, EcNR2.dnaG.Q576A, Nuc5-, and Nuc5-.dnaG.Q576A are presented here. As such, significance was defined as $P < 0.003$ for the analyses presented in Figures 3-4 and 3-6. Statistical significance in Figures 3-4 and 3-6 are denoted using a star system where * denotes $P < 0.003$, ** denotes $P < 0.001$, and *** denotes $P < 0.0001$. In the case of the experiment comparing EcNR2 and EcNR2.dnaG.Q576A using leading targeting oligos (Figure S3-1), we tested for statistical significance using a single t-test with significance defined as $P < 0.05$.

For the experiment in which 10 oligos were targeted within *lacZ*, recombinants were identified by blue/white screening. The frequency of clones with 1 or more alleles replaced (# of white colonies / total # of colonies) was determined for every replicate. For white colonies only, a portion of *lacZ* gene was amplified with primers lacZ_jackpot_seq-f and lacZ_jackpot_seq-r (Table S3-1), using KAPA HiFi HotStart ReadyMix as described above. PCR purified (Qiagen PCR purification kit) amplicons were submitted to Genewiz for Sanger sequencing in both directions using either lacZ_jackpot_seq-f or lacZ_jackpot_seq-r. Combined, the two sequencing

reads for each clone interrogated all 10 alleles (*i.e.*, unmodified or mutant sequence). Three replicates of recombinations and blue/white analysis were performed to ensure consistency, but only one replicate was sequenced (n = 39 for EcNR2 and n = 55 for EcNR2.dnaG.Q576A). Mean number of alleles replaced per clone were determined as described above. We tested for statistically significant differences in mean allele conversion between the strains using a Student's t-test with significance defined as $P < 0.05$. Statistical significance in Figure 3-5C is denoted using a star system where *** denotes $P < 0.0001$.

Supplemental material

Supplemental material for CHAPTER 3 can be found in APPENDIX B or at <http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0044638#s5> and <http://nar.oxfordjournals.org/content/suppl/2012/07/26/gks751.DC1/nar-01176-met-k-2012-File007.pdf>.

References

1. H. H. Wang *et al.*, Programming cells by multiplex genome engineering and accelerated evolution. *Nature* **460**, 894 (Aug, 2009).

2. H. H. Wang *et al.*, Genome-scale promoter engineering by coselection MAGE. *Nat. Methods* **9**, 591 (Jun, 2012).

3. P. A. Carr *et al.*, Enhanced multiplex genome engineering through co-operative oligonucleotide co-selection. *Nucleic Acids Res.*, (May 25, 2012, 2012).

4. H. O. Smith, C. A. Hutchison, C. Pfannkoch, J. C. Venter, Generating a synthetic genome by whole genome assembly: phi X174 bacteriophage from synthetic oligonucleotides. *Proc. Natl. Acad. Sci. U. S. A.* **100**, 15440 (Dec, 2003).

5. D. G. Gibson *et al.*, Creation of a Bacterial Cell Controlled by a Chemically Synthesized Genome. *Science* **329**, 52 (Jul, 2010).

6. H. M. Ellis, D. G. Yu, T. DiTizio, D. L. Court, High efficiency mutagenesis, repair, and engineering of chromosomal DNA using single-stranded oligonucleotides. *Proc. Natl. Acad. Sci. U. S. A.* **98**, 6742 (Jun, 2001).

7. F. J. Isaacs *et al.*, Precise Manipulation of Chromosomes in Vivo Enables Genome-Wide Codon Replacement. *Science* **333**, 348 (Jul, 2011).

8. X. T. Li *et al.*, Identification of factors influencing strand bias in oligonucleotide-mediated recombination in Escherichia coli. *Nucleic Acids Res* **31**, 6674 (Nov 15, 2003).

9. H. H. Wang, G. Xu, A. J. Vonner, G. M. Church, Modified bases enable high-efficiency oligonucleotide-mediated allelic replacement via mismatch repair evasion. *Nucleic Acids Res* **39**, 7336 (Sep 1, 2011).

10. X. T. Li *et al.*, Identification of factors influencing strand bias in oligonucleotide-mediated recombination in Escherichia coli. *Nucleic Acids Res.* **31**, 6674 (Nov, 2003).

11. H. H. Wang *et al.*, Genome-scale promoter engineering by coselection MAGE. *Nat Meth* **9**, 591 (2012).

12. J. A. Mosberg, M. J. Lajoie, G. M. Church, Lambda Red Recombineering in Escherichia coli Occurs Through a Fully Single-Stranded Intermediate. *Genetics* **186**, 791 (Nov, 2010).

13. M. Maresca *et al.*, Single-stranded heteroduplex intermediates in lambda Red homologous recombination. *BMC Mol Biol* **11**, 54 (Jul 29, 2010).

14. J. A. Sawitzke *et al.*, Probing cellular processes with oligo-mediated recombination and using the knowledge gained to optimize recombineering. *J Mol Biol* **407**, 45 (Mar 18, 2011).

15. H. H. Wang, G. Xu, A. J. Vonner, G. Church, Modified bases enable high-efficiency oligonucleotide-mediated allelic replacement via mismatch repair evasion. *Nucleic Acids Res* **39**, 7336 (Sep 1, 2011).

16. E. L. Zechner, C. A. Wu, K. J. Marians, Coordinated leading- and lagging-strand synthesis at the Escherichia coli DNA replication fork. II. Frequency of primer synthesis and efficiency of primer utilization control Okazaki fragment size. *Journal of Biological Chemistry* **267**, 4045 (February 25, 1992, 1992).

17. K. Tougu, K. J. Marians, The Extreme C Terminus of Primase Is Required for Interaction with DnaB at the Replication Fork. *J. Biol. Chem.* **271**, 21391 (August 30, 1996, 1996).

18. K. Tougu, K. J. Marians, The Interaction between Helicase and Primase Sets the Replication Fork Clock. *Journal of Biological Chemistry* **271**, 21398 (August 30, 1996, 1996).

19. J. W. Little, An exonuclease induced by bacteriophage lambda. II. Nature of the enzymatic reaction. *J Biol Chem* **242**, 679 (Feb 25, 1967).

20. M. Maresca *et al.*, Single-stranded heteroduplex intermediates in lambda Red homologous recombination. *BMC Mol. Biol.* **11**, (Jul, 2010).

21. A. Erler *et al.*, Conformational Adaptability of Red beta during DNA Annealing and Implications for Its Structural Relationship with Rad52. *J. Mol. Biol.* **391**, 586 (Aug, 2009).

22. G. Posfai *et al.*, Emergent properties of reduced-genome Escherichia coli. *Science* **312**, 1044 (May, 2006).

23. I. C. Blomfield, V. Vaughn, R. F. Rest, B. I. Eisenstein, Allelic exchange in Escherichia coli using the Bacillus subtilis sacB gene and a temperature-sensitive pSC101 replicon. *Mol. Microbiol.* **5**, 1447 (Jun, 1991).

24. S. Warming, N. Costantino, D. L. Court, N. A. Jenkins, N. G. Copeland, Simple and highly efficient BAC recombineering using galK selection. *Nucleic Acids Res.* **33**, e36 (January 1, 2005, 2005).

25. J. A. DeVito, Recombineering with tolC as a selectable/counter-selectable marker: remodeling the rRNA operons of Escherichia coli. *Nucleic Acids Res* **36**, e4 (Jan, 2008).

26. Y. Tashiro, H. Fukutomi, K. Terakubo, K. Saito, D. Umeno, A nucleoside kinase as a dual selector for genetic switches and circuits. *Nucleic Acids Res.* **39**, e12 (February 1, 2011, 2011).

27. A. J. Oakley *et al.*, Crystal and Solution Structures of the Helicase-binding Domain of Escherichia coli Primase. *Journal of Biological Chemistry* **280**, 11495 (March 25, 2005, 2005).

28. J. E. Corn, J. M. Berger, Regulation of bacterial priming and daughter strand synthesis through helicase-primase interactions. *Nucleic Acids Res.* **34**, 4082 (Sep, 2006).

29. G. Lia, B. Michel, J.-F. Allemand, Polymerase Exchange During Okazaki Fragment Synthesis Observed in Living Cells. *Science* **335**, 328 (January 20, 2012, 2012).

30. N. A. Tanner *et al.*, Single-molecule studies of fork dynamics in Escherichia coli DNA replication. *Nat Struct Mol Biol* **15**, 170 (2008).

31. M. Nakayama, O. Ohara, Improvement of recombination efficiency by mutation of Red proteins. *Biotechniques* **38**, 917 (Jun, 2005).

32. N. Y. Yao, R. E. Georgescu, J. Finkelstein, M. E. O'Donnell, Single-molecule analysis reveals that the lagging strand increases replisome processivity but slows replication fork progression. *Proceedings of the National Academy of Sciences* **106**, 13236 (August 11, 2009, 2009).

33. N. Rybalchenko, E. I. Golub, B. Bi, C. M. Radding, Strand invasion promoted by recombination protein β of coliphage λ. *Proc. Natl. Acad. Sci. U. S. A.* **101**, 17056 (December 7, 2004, 2004).

34. T. Asai, T. Kogoma, D-loops and R-loops: alternative mechanisms for the initiation of chromosome replication in Escherichia coli. *Journal of Bacteriology* **176**, 1807 (April 1, 1994, 1994).

35. H. H. Wang, G. M. Church, Multiplexed genome engineering and genotyping methods applications for synthetic biology and metabolic engineering. *Methods Enzymol* **498**, 409 (2011).

36. J. F. Jekel, D. L. Katz, J. G. Elmore, D. Wild, *Epidemiology, Biostatistics, & Preventative Medicine*. (W.B. Saunders, 2001).

37. N. Costantino, D. L. Court, Enhanced levels of lambda red-mediated recombinants in mismatch repair mutants. *Proc. Natl. Acad. Sci. U. S. A.* **100**, 15748 (Dec, 2003).

CHAPTER 4

Genome-wide Codon Replacement Using Synthetic Oligonucleotides and Engineered Conjugation

This chapter is reproduced with permission from its initial publication and with minor corrections:

Isaacs FJ[*], Carr PA[*], Wang HH[*], **Lajoie MJ**, Sterling B, Kraal L, Tolonen AC, Gianoulis TA, Goodman DB, Reppas NB, Emig CJ, Bang D, Hwang SJ, Jewett MC, Jacobson JM, Church GM (2011) *Genome-wide Codon Replacement Using Synthetic Oligonucleotides and Engineered Conjugation*. **Science**: 333, 348-353.

For clarity and consistency, all mentions of TAG and TAA codons have been changed to UAG and UAA.

Research contributions: F. Isaacs, P. Carr, H. Wang, and G. Church conceived of the project. F. Isaacs, P. Carr, H. Wang, M. Lajoie, B. Sterling, and L. Kraal converted UAG codons to UAA using MAGE. F. Isaacs, M. Lajoie, H. Wang, L. Kraal, and A. Tolonen performed the CAGE assemblies. F. Isaacs, M. Lajoie, T. Gianoulis, and D. Goodman performed the genome sequencing analysis. F. Isaacs, P. Carr, H. Wang, M. Lajoie, A. Tolonen, and N. Reppas helped develop the methods. F. Isaacs, P. Carr, H. Wang, M. Lajoie, J. Jacobson, and G. Church planned the experiments. F. Isaacs wrote most of the paper with help from M. Lajoie. All authors reviewed and edited the manuscript. F. Isaacs, P. Carr, H. Wang, M. Lajoie, J. Jacobson, and G. Church oversaw all aspects of the project.

Acknowledgements: We thank S. Zhang for helpful discussions and extensive use of laboratory resources, R. Kolter for the JC411 strain, C. and J. Seidman for Covaris E210, M. Sommer and L. Yang for sequencing advice, and members of the Church and Jacobson labs for helpful discussions. Supported by the NSF (SynBERC, Center for Bits and Atoms, and Genes and Genomes Systems Cluster), the U.S. Department of Energy, an NSF graduate fellowship (H.H.W.), a National Defense Science and Engineering Graduate Fellowship (M.J.L.), and an NIH K99/R00 award (M.C.J.).

Abstract

We present genome engineering technologies that are capable of fundamentally reengineering genomes from the nucleotide to the megabase scale. We used multiplex automated genome engineering (MAGE) to site-specifically replace all 314 UAG stop codons with synonymous UAA codons in parallel across 32 *Escherichia coli* strains. This approach allowed us to measure individual recombination frequencies, confirm viability for each modification, and identify associated phenotypes. We developed hierarchical conjugative assembly genome engineering (CAGE) to merge these sets of codon modifications into genomes with 80 precise changes, which demonstrate that these synonymous codon substitutions can be combined into higher-order strains without synthetic lethal effects. Our methods treat the chromosome as both an editable and an evolvable template, permitting the exploration of vast genetic landscapes.

Introduction

The conservation of the genetic code, with minor exceptions (*1*), enables exchange of gene function among species, with viruses, and across ecosystems. Experiments involving fundamental changes to the genetic code could substantially enhance our understanding of the origins of the canonical code and could reveal new subtleties of how genetic information is encoded and exchanged (*1, 2*). Modifying the canonical genetic code could also lead to orthogonal biological systems with new properties. For instance, a new genetic code could prevent the correct translation of exogenous genetic material and lead to the creation of virus-resistant organisms. Additionally, a recoded genome could enhance the incorporation of unnatural amino acids into proteins, because existing suppressor systems must compete with native translation factors (*3-5*).

The construction of a new genetic code requires methods to manipulate living organisms at a whole-genome scale. Such methods are only now becoming attainable through the advent of advanced tools for synthesizing, manipulating, and recombining DNA (*6*). This has led to a number of impressive genome-scale studies, which include removing transposable elements (*7*), refactoring phage genomes (*8*), genome merging (*9*), whole-genome synthesis (*10*), and transplantation (*11*). Whole-genome *de novo* synthesis offers the ability to create new genomes without a physical template. Its main limitations are the cost of accurate *in vitro* DNA assembly and introduction of synthetic DNA into organisms (*12*). For this reason, *de novo* synthesis is chosen when trying to create a small number of new DNA constructs of modest size (<10 kb) and high fidelity (*8, 10, 12, 13*). Notably, however, the digital template used in *de novo* synthesis currently originates almost exclusively from sequences found in nature or minor variants thereof.

Redesigned genomes require approaches that reconcile the desired biological behavior with challenges inherent to biological complexity. Engineering biological systems can be unpredictable, as a single misplaced or misdesigned allele can be lethal. To address these challenges, we have developed approaches that integrate synthetic DNA and recombination methods to introduce genome-wide changes dynamically in living cells, thereby engineering the genome through viable intermediates. In recent work, we developed multiplex automated genome engineering (MAGE), which rapidly generates genetic diversity for strain and pathway engineering (*14*). To augment MAGE's ability to introduce nucleotidescale mutations across the genome, a complementary method was required to assemble modified chromosomes *in vivo*.

Here, we report the development of a hierarchical conjugative assembly genome engineering (CAGE) method and its integration with MAGE toward reengineering the canonical genetic code of *E. coli* (Figure 4-1) – an organism with broad utility in basic and applied research. The *E. coli* genetic code contains three stop codons (UAG, UAA, and UGA) whose translation termination is mediated by two release factors, RF1 and RF2. RF1 recognizes the termination codons UAA and UAG, whereas RF2 recognizes UAA and UGA. We hypothesized that replacing all UAG codons with synonymous UAA codons would abolish genetic dependence on RF1 and permit the newly reassigned UAA codons to be recognized by RF2. This will enable us to test and leverage the redundancy of the genetic code by deleting RF1 (Δ*prfA*), providing a blank UAG codon that could be cleanly reassigned to new function. Given that codon utilization bias has been shown to affect translation efficiency (*15, 16*) and viral infectivity (*13*), we sought to determine whether *E. coli* could maintain viability with the systematic replacement of the 314 UAG codons.

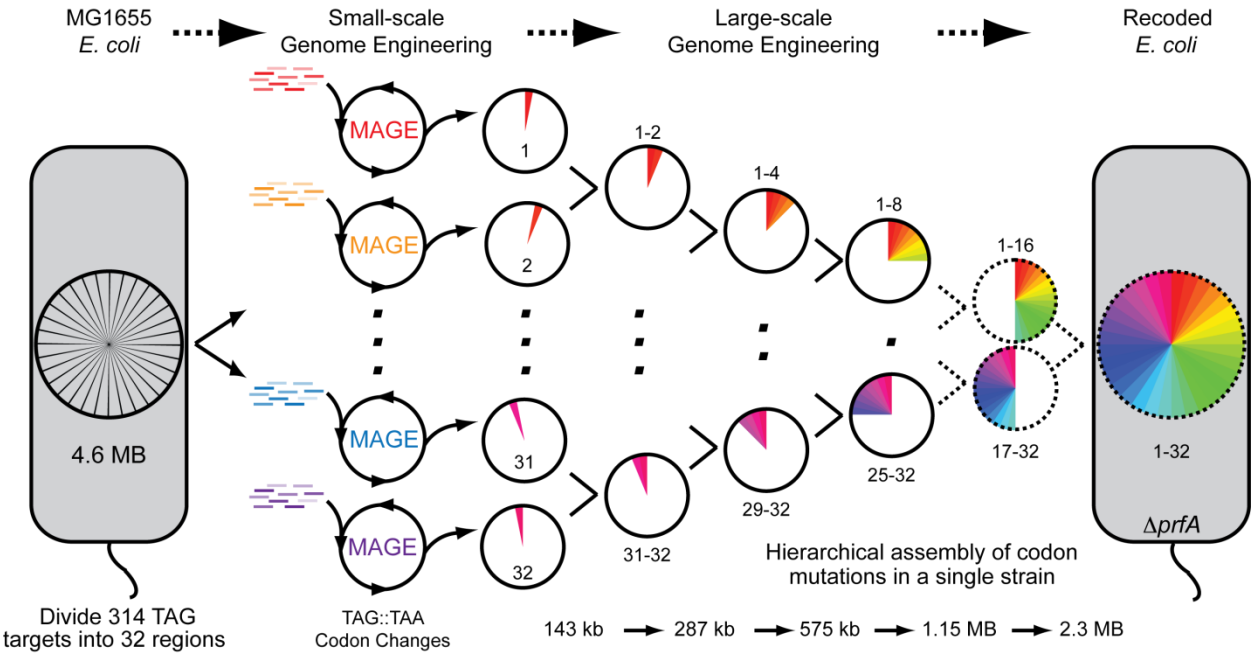


Figure 4-1. Strategy for reassigning all 314 UAG codons to UAA in *E. coli*. First, the genome was split into 32 regions each containing 10 UAG stop codons. In parallel, MAGE was used to execute all 10 UAG::UAA codon modifications in a single strain for each genomic region. These partially recoded strains were paired such that a targeted genomic region of one strain (donor) was strategically transferred into a second strain (recipient), permitting the hierarchical consolidation of modified genomic regions using CAGE (see Figure 4-4A). This five-stage process transfers genomic fragments ranging in size from ~154 kb to ~2.3 Mb in a controlled manner until a single recoded strain is constructed that lacks the UAG stop codon throughout. Thus far, 28 of 31 conjugations have been completed, where the dotted arrows denote outstanding conjugation steps and dotted genomes represent half- and full-genome strains that have not yet been completed. Once all UAG codons have been converted to UAA, the *prfA* gene will be deleted to inactivate UAG translational termination.

Results and Discussion

On the basis of the MG1655 genome annotation, we identified 314 *E. coli* genes that contain the UAG stop codon (Figure S4-1 and Table S4-1). We focused initially on reassigning all 314 stop codons (UAG) to the synonymous stop codon (UAA) in a modified *E. coli* MG1655 strain {EcNR2: *E. coli* MG1655 Δ(*ybhB-bioAB*)::[λ *cI857* Δ(*cro-ea59*)::tet^R-*bla*] Δ*mutS*::*cat*}. A mismatch repair deficient (Δ*mutS*) strain was used to achieve high-frequency allelic replacement (*17*). We used MAGE to simultaneously introduce subsets of the UAG-to-UAA codon changes into 32 separate strains (Figure 4-1). Specifically, the EcNR2 genome was divided into 32 regions; 31 of these contained 10 targets, and the other contained the remaining four targets. This division was pursued for four reasons. First, pilot experiments (Figure S4-2) and associated

computational predictions showed that the use of pools of 10 or more oligonucleotides (oligos) for MAGE (*14*) achieves highly efficient allelic replacement. Second, limiting the number of MAGE cycles for codon conversions minimizes the total number of cell divisions (six to eight per cycle) in the presence of λ red proteins (which promote recombination and are mutagenic) and deficient mismatch repair (MMR) (*18*). This reduces the number of undesired secondary mutations. Third, the use of smaller oligo pools enabled rapid accumulation of the desired codon conversions in parallel and quantitative measurements of conversion frequencies. Finally, we anticipated that certain codons might be recalcitrant to codon conversion or cause an aberrant phenotype, so it was advantageous to test mutations in small subsets. Candidates included 43 essential genes (*19*) that are terminated by UAG (Figure S4-1) and 39 genes in which the UAG stop codon overlaps a second reading frame (Table S4-2). Thus, parallel allelic replacement across the 32 regions in separate strains would enable rapid identification of potentially troublesome alleles.

The 314 oligonucleotides encoding the specified UAG-to-UAA codon mutations (Table S4-3) were computationally designed by means of a software tool (optMAGE, http://arep.med.harvard.edu/optMAGE) on the basis of prior MAGE optimization experiments (*14*). These oligos were repeatedly applied over 18 MAGE cycles to introduce the codon replacements across 32 cultures (10 targets per culture). We developed two methods based on mismatch amplification mutation assay polymerase chain reaction (MAMA-PCR) (*20*) to quickly assay target codons. Multiplex allele specific colony quantitative PCR (MASC-qPCR) (*21*) (Figure S4-3) was used to identify clones that contain the greatest number of codon conversions, and multiplex allele specific colony PCR (MASC-PCR) (*21*) (Figure S4-4) was used to measure frequencies of allele replacement at each targeted position. MASC-PCR permitted simultaneous

single–base pair (1 bp) measurements (UAG versus UAA) at 10 chromosomal sites per clone (Figure S4-4).

After 18 MAGE cycles, allelic replacement frequencies were analyzed for all 314 UAG-to-UAA mutations (Figure 4-2) in 1504 clones (47 clones for each of the 32 recoded segments). Allelic replacement frequencies exhibited a high degree of variability among the targets (Figure 4-2, outer ring; Table S4-4). The average allelic replacement frequency observed was 37 ± 19% after 18 cycles, and 42% of the population was unconverted after 18 cycles; we observed 1 to 10 converted alleles per clone across the remaining population (Figure 4-3A). These measurements suggest the evolution of two types of cells in our mixed cultures: one that appears largely resistant to allelic replacements, and another that readily permits them. With this knowledge, future MAGE methodology could be modified to select highly recombinogenic clones after fewer cycles (*e.g.*, 5 versus 18). Notably, comparable distributions of allelic replacement frequencies were observed for UAG codons present in essential genes, codons overlapping a second reading frame, and codons distributed at various positions throughout both replicating arms (Table S4-5). Moreover,

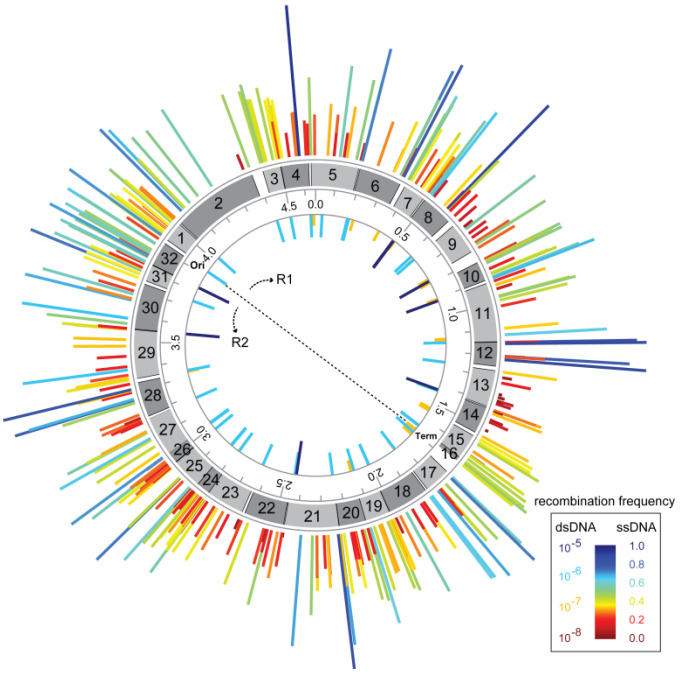


Figure 4-2. Frequency map of oligo-mediated UAG::UAA codon replacements and genetic marker integrations across the *E. coli* genome at each replacement position. Circular map illustrates (from inner circle outward): (i) frequency of dsDNA selectable marker integrations; (ii) genome coordinates (in Mb): position of origin (Ori) and terminus (Term) and direction of the two replication forks (R1 and R2); (iii) location of the 32 targeted chromosomal segments; and (iv) frequency of UAG::UAA replacements across all UAG codons—after 18 MAGE cycles—denoted by height- and color-coded bars (scale bar indicates integration frequency).

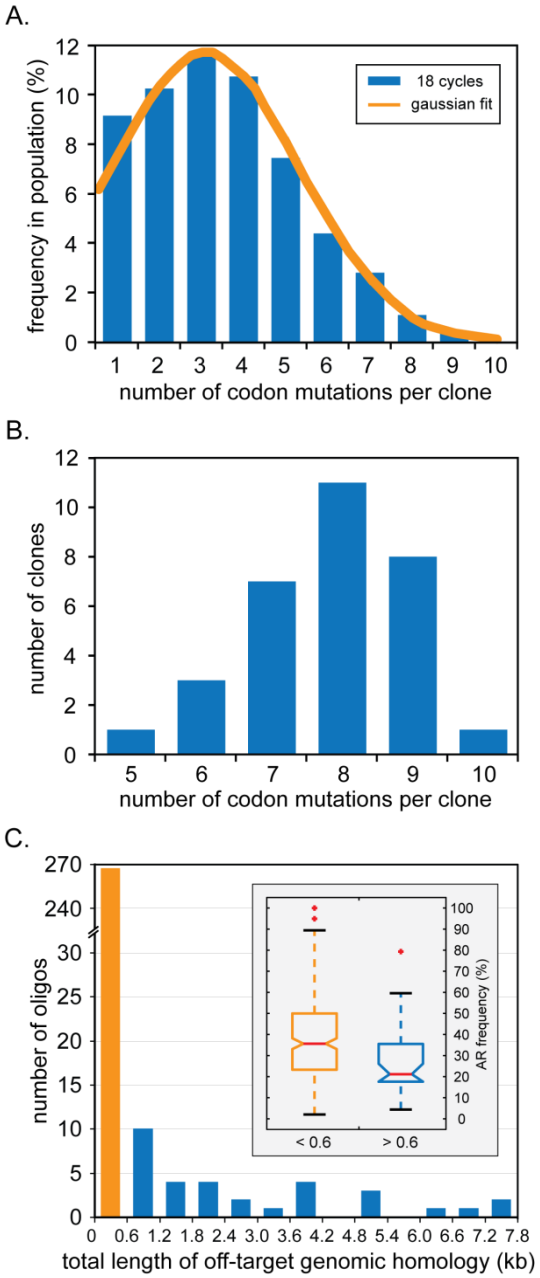


Figure 4-3. Clonal rate and distribution of genome modifications after 18 cycles of MAGE. **(A)** Histogram of the frequencies of clones containing 1 to 10 conversions found among 1504 clones screened. A Poisson fit is shown (solid yellow line) for a subpopulation that excludes the zero-conversion group. **(B)** Distribution of modifications among the group of top clones (one for each of the 31 groups of 10 targeted modifications; one additional strain not shown had conversion at all four codon sites). **(C)** Distribution of the 314 90-mer oligos by their extent of total secondary sequence similarity to the *E. coli* genome. Inset box plot: Oligos with a mistarget score of more than 600 bp show, on average, a 32% decrease in allelic replacement (AR) frequency relative to oligos scoring less than 600 bp (25.6% versus 37.6%, $P < 0.003$).

allelic replacement frequencies did not correlate with distance from the origin of replication (*oriC*), nor with recombination hotspots [*e.g.*, Chi-sites, DNA motifs (5′-GCTGGTGG-3′) in the genome where homologous recombination could be enhanced] or direction and level of transcription.

All individual UAG-to-UAA conversions were observed, indicating that no UAG stop codon in *E. coli* is required for survival or robust growth. Of 314 codon targets, 298 could be assayed using MASC-PCR, whereas the remaining modifications were confirmed by direct DNA sequencing. By screening to identify maximally modified cells (Figure 4-3B, 5 to 10 modifications per clone with a median of 8) and minimizing aberrant phenotypes (*i.e.*, auxotrophy, decreased fitness) across 1504 clones, we isolated the top clone from each of the 32 populations after 18 MAGE cycles. These clones collectively accumulated 246 of 314 (78%) desired mutations after 18 MAGE cycles. Clones that did not contain all of the

codon changes were subjected to an additional 6 to 15 MAGE cycles to convert the remaining UAG codons.

Given that λ Red facilitates highly efficient recombination using short regions of complementary sequences, it was important to assess the potential effects of oligonucleotide hybridization to other (unintended) regions of the genome. We found that 90-mer oligos that have multiple regions of high sequence similarity throughout the genome have a reduced recombination frequency (Figure 4-3C) but that these oligos rarely cause mutagenesis at those other locations (see below). To estimate this effect quantitatively, we performed BLAST alignments of each oligo against the entire genome. To compute a mistarget score, we summed the lengths of the BLAST matches for each oligo sequence against the rest of the *E. coli* genome (blastn, word size = 11, expectation value =10). Although the majority of oligos (~270) showed only minor sequence similarity to the genome (mistarget score < 600), we found that the score strongly correlated with the frequency of allelic replacement (Figure 4-3C). Recombination frequencies were decreased by more than 30% for oligos having many regions of high sequence similarity in the genome (mistarget scores > 600 bp; $P < 0.003$). This information will be useful as a predictor of allele replacement frequency for future oligo designs and can be incorporated into automated design software such as optMAGE.

To directly verify the presence of codon conversions and to obtain a snapshot of secondary mutations accumulated during the MAGE process, we performed Sanger sequencing of ~300 bp surrounding each modified UAG replacement site (96 kb overall, ~3 kb in each of the 32 top strains). Sequencing confirmed the accuracy of our MASC-PCR method and verified the 16 UAG-to-UAA conversions not detected by this assay. Background mutations outside of the 90 bp regions targeted by MAGE oligos consisted of 6 substitutions, 0 insertions, and 3

deletions; in contrast, mutations within the targeted regions included 4 substitutions, 1 insertion, and 28 deletions (Figure S4-5). The use of a MMR-deficient (Δ*mutS*) strain rendered the expected bias toward substitution mutations in the nontargeted regions. Deletion mutations are probably enriched in oligo-targeted regions because internal deletions are common errors in many oligonucleotide chemical synthesis processes (*22, 23*). We have developed strategies to minimize both sources of error: (i) optimized oligo synthesis to reduce deletions (*e.g.*, Figure S4-7), and (ii) the use of chemically modified oligos that are not recognized by MMR to achieve efficient allelic replacement in the presence of a functional MMR pathway.

Because we initially performed codon changes in small subsets, we could easily identify candidate mutations that lead to aberrant phenotypes. Growth rates across all 32 top strains (Figure S4-6 and Table S4-6, average of 47 min per doubling) showed modest deviations from the growth rate of the ancestral strain (42 min per doubling). These changes in growth could be attributed to either the codon changes or the accumulation of secondary mutations in our MMR-deficient strain. Additional phenotypic assays showed a sustained high recombination frequency and a 2.8% frequency of auxotrophy on minimal M9–glucose minimal medium after ~366 generations (Table S4-6). These values compare favorably to previous studies (*24*) in which serial passage of a Δ*mutS* strain resulted in 9% frequency of auxotrophy after ~250 generations.

After converting all UAG codons to UAA across 32 *E. coli* strains, we initiated a five-stage hierarchical assembly (Figure 4-1 and Table S4-7) of the modified chromosomal segments into a single strain (Figure 4-4). To accomplish this, we developed the hierarchical CAGE method, which is rooted in conjugation, a key mechanism for gene transfer in bacteria (*25, 26*). In contrast to natural mechanisms of conjugal DNA transfer where the *oriT* sequence and conjugal factors act as a contiguous genetic construct, our approach physically decouples the

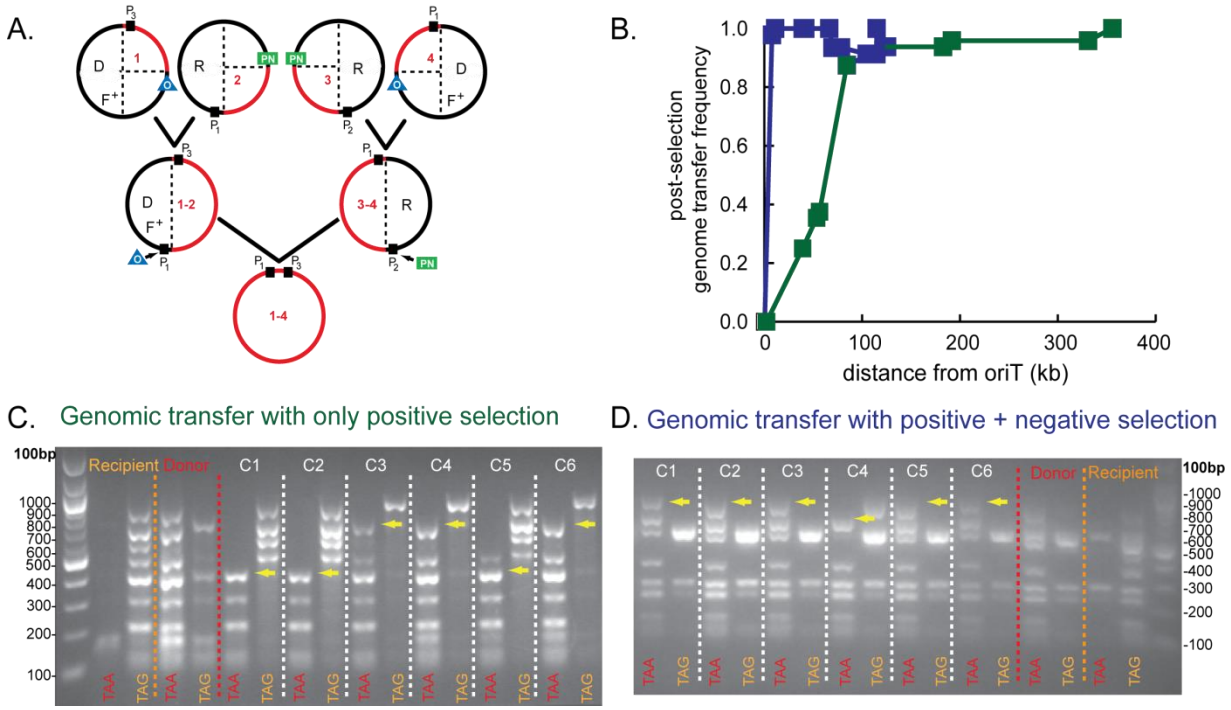


Figure 4-4. Hierarchical CAGE methodology for controlled genome transfer. **(A)** Two pairs of strains illustrate the design and methodology of CAGE, with recoded genomic regions in red. Partially recoded strains are split into conjugation pairs. The donor strain (D) contains the following: *oriT-kan* cassette (O, blue triangle); positive selection gene (P_n, n = 1, 2, or 3, black rectangle); and pRK24 (F^+). The recipient strain (R) contains the following: positive-negative selection gene (PN) and P_n, flanking its recoded region. DNA transfer is initiated at O in the donor genome, ensuring transfer of the desired codon mutations and downstream P_n. After conjugation, a specific set of three simultaneous selections is applied to yield a recombinant strain that contains the recoded genomic fragment from the donor strain while retaining the other recoded region in the recipient genome. Placement of the PN marker downstream of the *oriT* sequence in the recipient genome ensures that the entire desired region of the donor genome is inherited in the recombinant strain. All conjugation factors are maintained episomally on pRK24, so only a ~2 kb *oriT-kan* sequence must be inserted onto the genome to generate a highly controllable Hfr donor strain. Because there is no scar between the two recoded regions from the conjugation parents, only one recombination is required to insert the O (donor) or PN (recipient) directly into P_n for the next round of conjugation. This conjugation assembly selection strategy is implemented in five stages to merge the genomes of 32 recoded strains into a single strain (see Figure 4-1). **(B)** Genome transfer frequency as a function of the distance from O. Plots of two conjugations of genome segments illustrating the transfer of ~120 kb with positive and negative selection (blue) and ~360 kb with only positive selection (green) to assemble recoded genomic DNA from donor and recipient genomes. **(C and D)** MASC-PCR images of UAA alleles transferred under positive selection alone **(C)** and positive and positive-negative selection **(D)**. Yellow arrows indicate the genomic point of transfer, which illustrate the inheritance of the donor UAA alleles in the conjugated strain. UAG and UAA codons are assayed with primers that utilize 3' TAG and TAA DNA sequences, respectively.

episomally expressed conjugal factors from the chromosomally integrated *oriT* sequence. The *oriT* sequence is fused with a kanamycin resistance gene (*oriT-kan*) so that it can be easily integrated into any permissible locus across the *E. coli* genome *via* λ Red–mediated double-stranded DNA (dsDNA) recombination (*27*). Thus, we can precisely control the genomic

position at which conjugal transfer is initiated (Figure 4-4A). This strategy allowed us to use a tractable ~2 kb cassette in place of a cumbersome 30 kb Hfr fragment for consecutive manipulations throughout the genome.

Before conjugation, we converted the 32 strains that collectively contain all UAG-to-UAA modifications into 16 pairs of strains primed for large-scale genome transfer (Figure 4-1). Within each conjugation pair, a donor strain transfers its recoded genomic region to a recipient strain, which inherits the donor genome and retains its recoded genomic region. Genome transfer is controlled by the precise placement of positive and positive-negative selectable markers integrated with an engineered conjugation strategy to obtain the desired recombinant genomes. Precise placement of these markers into "safe insertion regions" (SIRs: intergenic regions that are not annotated for any coding or regulatory function) by dsDNA recombination (*27*) was intended to maintain genomic integrity and to attain the desired combination of recoded donor and recipient genomes in the recombinant strain (Figure 4-4A). In total, two genetic markers were inserted into each of the donor and recipient strains, yielding a total of 64 markers across the 32 modified strains. In the donor strain, the recoded region was flanked by an upstream *oriT-kan* cassette and a downstream positive selectable marker (P1, *e.g.*, zeo^R, $spec^R$, $gent^R$). In the recipient strain, the recoded region was flanked with a different positive selectable marker (P2) and a positive-negative selectable marker (PN) such as *tolC* (*28*) or *galK* (*29*). The frequencies of integration among selectable marker cassettes exhibited a high degree of site-specific variability (Figure 4-2, inner ring). On average, 59 clones (~10^{-6} frequency) were observed per recombination. However, dsDNA recombination frequencies spanned >3 logs across 81 integration sites tested. Twelve intergenic sites yielded no observable recombinants despite

repeated (three or more) attempts. The remaining 69 sites performed as follows: 23 sites at ~10^{-7}, 38 sites at ~10^{-6}, and 8 sites at ~10^{-5} recombination frequencies.

Placement of complementary selectable markers across all 32 strains served as anchor points that enabled hierarchical assembly of recoded genomic fragments. By design, this permits the use of modular *oriT-kan* and *tolC* cassettes throughout the assembly process. Rather than having to prepare a cassette for each SIR, three *oriT-kan* cassettes and three *tolC* cassettes that insert directly into the three positive markers (*zeo*^{R}, *spec*^{R}, and *gent*^{R} genes) are sufficient to guide the remaining four stages of hierarchical assembly. Because *oriT-kan* and *tolC* are not inherited by the recipient strain, each strain can be prepared for subsequent conjugations by simply inserting an *oriT-kan* (donor) or a PN (recipient) directly into one of the strain's inherited positive markers (Figure 4-4A).

In the first stage of the hierarchical conjugation strategy (Figure 4-1 and Table S4-7), 32 strains each containing 10 codon modifications were merged to produce 16 strains with 20 modifications. Transfer of 1/32 of the genome (~150 kb) occurs at a frequency of ~10^{-4} (Figure 4-4B), 2 logs greater than half-genome transfer (*21*). This result supports prior findings that the probability of transferring a specific marker decreases exponentially with its distance from *oriT* (*26*). The relationship between genome transfer efficiency and the distance from *oriT* revealed useful parameters for designing our engineered conjugation scheme. In the absence of a positive-negativemarker in the recipient strain, MASC-PCR analysis showed reduced transfer frequency from loci that are in close proximity (<10 kb) to *oriT*, resulting in the uncontrolled transfer of the donor genome—specifically, the loss of mutated UAA codons from the donor genome and the retention of one to four UAG codons from the recipient genome (Figure 4-4C). Upon the inclusion of a positive-negative marker [*e.g.*, *tolC* (*28*), *galK (29)*] in the recipient genome,

desired postconjugal strains were readily selected; that is, full transfer of mutated UAA codons from the donor genome was achieved by selecting for the loss of *tolC* or *galK* placed among the UAG codons in the recipient genome (Figure 4-4D). Together, these results demonstrate the requirement for robust positive and positive-negative selectable markers that strategically flank the recoded genomes in the donor and recipient strains (Figure 4-4A). Moreover, MASC-PCR analysis across all codon loci shows that conjugation efficiency is sustained throughout the region of transfer, indicating contiguous transfer of the donor genomic fragment.

Using CAGE, we then consolidated the 32 original strains into eight recoded strains, each with 1/8 of the genome recoded. Two of these eight strains exhibited a dysfunctional *tolC* phenotype (*i.e.*, they simultaneously passed both positive and negative *tolC* selections). Although mutations conferring simultaneous novobiocin sensitivity and colicin E1 resistance have been identified (*28*), there is no literature precedence for the phenotype that we observed. We have discovered two routes to this phenotype. In one strain, the causative allele was present in *tolC*, and we corrected the phenotype by replacing the dysfunctional copy with a functional one. In the other strain, the causative allele appears to be outside of *tolC*. Indeed, *tolC* works in concert with a number of other genes (*e.g.*, *btuB*, *tolA*, *tolQ*, and *tolR*) that have been implicated in dysfunctional negative selection (*30*). Recognizing that the ancestral strains also carried the dysfunctional allele, we reconstructed this1/8th strain using MAGE, and used it to complete the full set of 1/4 genomes (28 of 31 conjugations).These four strains, which contain up to 80 modifications per genome, can be combined to complete the assembly of a fully recoded strain containing all 314 UAG-to-UAA codon conversions.

In light of the challenges arising from spontaneous point mutations, we sought to assess the effects of MAGE and CAGE on genome stability. Therefore, we performed whole-genome

sequencing for the two dysfunctional strains and an additional functional control (Figures S4-8 and S4-9 and Tables S4-8 to S4-11). These strains have 110, 102, and 128 secondary mutations, respectively [total number of single-nucleotide polymorphisms (SNPs) and indels; Figure S4-8]. After ~960 cell divisions (Table S4-11), the majority of SNPs were transition mutations (98.4% transitions and 1.6% transversions; Table S4-9) and the overall background mutation rate was 2.5×10^{-8} per bp per replication (1 error per genome per ~9 replications; Table S4-10). These results are consistent with a Δ*mutS* phenotype (*31*). Our measured error rate was lower than we expected, given that the cumulative potential mutations include contributions from a MMR-deficient strain, repeated exposure to induction of the λ Red recombination system, and conjugation-based genomic manipulations. A mechanistic hypothesis for the lower error rate is that the conjugation process acts as a backcross and removes deleterious secondary mutations through the isolation of clones that maximize fitness. To examine this idea further, we explored the potential functional consequences of these SNPs as indicated by the COG category of the gene or regulatory region associated with the SNP (*32-34*).We used a hypergeometric distribution to determine the enrichment level of the three main COG categories across all three strains. Both SNPs associated with metabolism (117 SNPs, $P < 0.0004$) and SNPs associated with information storage and processing (29 SNPs, $P < 0.05$) were shown to be significantly enriched, whereas SNPs associated with cell signaling and transduction (98 SNPs, $P > 0.05$) were not (Figure S4-9). Future work will be needed to sequence additional strains throughout the ancestral conjugation tree to characterize the frequency, inheritance patterns, and functional bias of such mutations.

Discussion

This study, which integrates *in vivo* genome engineering from the nucleotide to the megabase scale, demonstrates the successful replacement of all genomic occurrences of the UAG stop codon in the *E. coli* genome. We found that cells can incorporate all individual UAG-to-UAA codon changes, and that these changes can be assembled into genomes with up to 80 modifications with mild phenotypic consequences. The scarless introduction of codon changes *via* MAGE enabled the first genome-wide allelic replacement frequency map using single-stranded DNA oligos in *E. coli* (Figure 4-2). In addition, our engineered conjugation experiments produced a complementary recombination frequency map of intergenic dsDNA integration sites across the genome (Figure 4-2). Together, these experiments revealed both highly accessible and recalcitrant sites for both small- and large-scale chromosomal modifications. These data could serve as valuable resources for future genome engineering efforts. Moreover, synthetic approaches such as the one pursued here may help to refine the existing genome annotation by revealing unannotated functional genetic loci, such as short peptides (*35*) or minigenes (*36*).

Introducing genome-wide changes dynamically in a living cell permits engineering in the cell's native biological context. In contrast to *in vitro* genome synthesis (*10*) and transplantation methods (*12*) that introduce discrete and abrupt changes in a single genome, our genome engineering technologies treat the chromosome as an editable and evolvable template and generate targeted and combinatorial modifications across many (~10^9) genomes *in vivo* (*14*). MAGE is optimal for introducing small modifications in sequence design space, whereas CAGE is designed for taking bigger leaps *via* large-scale assembly of many modified genomes. Together, these genome editing methods are advantageous when the designed genomes share >90% sequence similarity to existing templates or when many targeted mutations dispersed across the chromosome are desired (*e.g.*, genome recoding).

Materials and Methods

Strains and Culture Conditions: The λ prophage was obtained from strain DY330 (*27*), modified to include the *bla* gene and introduced into wild-type MG1655 E.coli by P1 transduction at the *bioA*/*bioB* gene locus and selected on ampicillin to yield the strain EcNR1 (λ Red^+). Replacement of *mutS* with the chloramphenicol resistance gene (chloramphenicol acetyl transferase, *cat*) in EcNR1 produced EcNR2 (*mutS*$^-$, λ Red^+). EcNR2 was grown in low salt LB-Lennox medium (LB^L; 10 g tryptone, 5 g yeast extract, 5 g NaCl in 1 L dH_2O) for optimal electroporation efficiency and compatibility with zeocin selection. EcNR2 was used as the ancestral strain for all recoded strains reported in this manuscript.

Oligonucleotides: All oligonucleotides were obtained from Integrated DNA Technologies. Oligonucleotides (Table S4-3) used in the MAGE process were designed according to the following specifications: 1) 90 nucleotides in length, 2) contain a single mutation to effect the UAG to UAA codon conversion, 3) two phosphorothioate linkages at both the 5' and 3' ends to attenuate exonuclease activity and to increase half-life, 4) minimize secondary structure (ΔG threshold values, self-folding energy), 5) target lagging strand at the replication fork. No additional purification was used following oligonucleotide synthesis. Primers were purchased from IDT for the multiplex PCR assays and loci sequencing reactions (see description below and Tables S4-12 and S4-13).

MAGE-generated Codon Conversions: A single clone of the EcNR2 strain was grown in liquid cell culture, which was used to inoculate 32 separate cultures for parallel modification of all UAG codons. Modification of these codons was achieved through continuous MAGE (*14*) cycling. Each culture was grown at 30°C to mid-logarithmic growth (*i.e.*, OD_{600} of ~0.7) in a

rotor drum at 200 RPM. To induce expression of the λ Red recombination proteins (Exo, Beta and Gam), cell cultures were shifted to a 42 °C water bath with vigorous shaking for 15 min and then immediately chilled on ice. In a 4 °C environment, 1 mL of cell culture was centrifuged at 16,000x g for 30 seconds. Supernatant media was removed and cells were re-suspended in 1 mL dH_2O (Gibco cat# 15230). This wash process was repeated. Supernatant water was removed, and the pellet was re-suspended in the appropriate pool of 10 oligos (1 uM per MAGE oligo in 50 uL dH_2O). The re-suspended oligos/cell mixture was transferred to a pre-chilled 96-well, 2 mm gap electroporation plate (BTX, USA) and electroporated with a BTX electroporation system using the following parameters: 2.5 kV, 200 Ω, and 25 μF. The electroporated cells were immediately transferred to 3 mL of LB^L media for recovery. Recovery cultures were grown at 30 °C in a rotator drum for 2-2.5 hours. Once cells reached mid-logarithmic growth they proceeded to the next MAGE cycle. This approach introduces genomic modifications while allowing cells to evolve and adapt to those changes. Moreover, this approach is designed to explore extensive genotype and phenotype landscapes by creating combinatorial genomic variants that leverage the size of the cell population. After 18 MAGE cycles, cells from each population were isolated on LB^L agar plates. Forty-seven clones from each of the 32 cycled populations were selected and subjected to genotype and phenotype analyses. From each population the clone with the greatest number of modifications (an average of 8 modifications per clone) and minimal aberrant phenotypes (*i.e.*, auxotrophy, decreased fitness) was selected. Further MAGE cycles were employed (typically 6 cycles, but in some cases up to 15) to yield strains with complete sets of 10 targeted modifications.

Genotype Analyses: UAG-to-UAA codon conversions were analyzed using three main methods: 1) Multiplex allele specific colony PCR (MASC-PCR), 2) Multiplex allele specific colony quantitative PCR (MASC-qPCR) and 3) Sanger DNA sequencing.

Multiplex Allele Specific Colony PCR (MASC-PCR): Based on previously described allele-specific PCR techniques, we developed the MASC-PCR method to test for UAG-to-UAA codon conversion in our recoded strains (the ancestral EcNR2 strain was the negative control). Three primers were designed for each locus: 1) a forward primer for the UAG sequence, 2) a forward primer for the UAA sequence and 3) a reverse primer compatible with both forward primers (Table S4-12). Primers were designed for a target Tm of 62 °C. The two forward primers were identical except that the most 3' nucleotide hybridized to produce either a GC base pair for the wildtype (UAG) codon or an AT base pair for the mutant (UAA) codon. Thus, every clone from each of the 32 populations was interrogated *via* two MASC-PCR reactions, in which each reaction assayed 10 different loci (with one set assaying four loci). One reaction assayed the wild-type (UAG) sequence and a second reaction assayed the mutant (UAA) sequence, yielding two binary reactions that revealed the sequences of the targeted codons (Figure S4-4). A clone containing the mutant allele generated PCR products only using the mutant allele primers and not the WT primers and *vice versa* for a clone with the wild-type allele. To minimize nonspecific amplification of MASC-PCR primers, a gradient PCR was performed to experimentally determine the optimal annealing temperature for each MASC-PCR primer pool (typically between 64 – 67 °C). Multiple loci were queried in a single PCR reaction using the multiplex PCR master mix kit from Qiagen. Each MASC-PCR primer set produced amplicon lengths of 100, 150, 200, 250, 300, 400, 500, 600, 700, or 850 bps, corresponding to up to 10 different genomic loci. We found that using a 1:50 dilution of saturated clonal culture in water as template

generated the best MASC-PCR specificity. Typical 20 uL MASC-PCR reactions included 10uL 2x Qiagen multiplex PCR master mix, 0.2 uM of each primer, and 1 uL of template. MASC-PCR cycles were conducted as follows: polymerase heat activation and cell lysis for 15 min at 95 °C, denaturing for 30 sec at 94 °C, annealing for 30 sec at experimentally determined optimal temperature (64 – 67 °C), extension for 80 sec at 72 °C, repeated cycling 26 times, and final extension for 5 min at 72 °C. Gel electrophoresis on a 1.5% agarose gel (0.5x TBE) produced the best separation for a 10-plex MASC-PCR reaction. (See Figure S4-4 for representative gel picture of MASC-PCR reaction).

Mulitplex allele-specific quantitative colony PCR (MASC-qPCR): In complement to MASC-PCR analyses, we also developed a highly multiplexed quantitative PCR screen to rapidly identify highly modified clones (Figure S4-3). Typical multiplexed qPCR reactions employ multiple fluorescence and distinct detection events to assess multiple PCR reactions in one sample, and are generally limited by the available optics and fluorescence to 4 channels. Instead, we needed a robust, economical test that employed many different non-optimized primers, did not require more expensive fluorescently labeled oligos, and would work for 10-plex reactions. We accomplished these goals with SYBR Green I detection, which gauges the total amount of DNA produced in the reaction. Two qPCR reactions were compared for each clone evaluated, one with 10 pairs of primers matched to the unmodified UAG genes, and the other with 10 primer pairs matched to the intended UAA modifications. The UAG reactions were expected to proceed most efficiently with a wild-type template, and the UAA reactions most efficiently with a fully modified template. Intermediate values between these extremes also provided an effective, though nonlinear gauge of the extent of modification for each clone (Figure S4-3A-C).

Each colony was used as template for a pair of qPCR reactions comparing the amplification efficiency when matched to primers terminating in wild-type or targeted mutant sequence. The experimental measurement for a given clone is then compared to the equivalent values measured for the unmodified starting (negative control) strain. This reference value is subtracted from each ΔC_t to yield a $\Delta\Delta C_t$, with unmodified clones scoring close to zero (as with the negative control colonies). The largest $\Delta\Delta C_t$ values were expected to indicate the most modified clones, which we confirmed by genotyping clones with varying $\Delta\Delta C_t$ values (Figure S4-3C). Large numbers of clones could be quickly assessed using this approach (up to 190 per 384-well plate, plus 2 negative controls). A typical assessment of MAGE-cycled clones comprised of 4 groups per plate, *i.e.*, for each culture targeting 10 modifications, 2-4 control colonies and 44-46 queried colonies. After identification of the most promising clones, site-specific qPCR genotyping (Figure S4-3D) was used to identify which specific sites had been modified, selecting the best clones for further modification.

Individual bacterial colonies were picked into 0.5 mL sterile distilled deionized water, with 5 μL of this suspension used as template in 20 μL qPCR reactions containing 1x NovaTaq buffer, 0.5 U NovaTaq Hotstart DNA Polymerase (EMD Biosciences), 250 μM each dNTP, 0.5x SYBR Green I (Invitrogen), and 5% DMSO. Primer concentrations were 50 nM for each primer (*i.e.*, 500 nM total for10 forward primers and 500 nM total for 10 reverse primers). A typical qPCR program included a 10 minute hot start at 95 °C, followed by 40 cycles (95 °C for 30 seconds, 60 °C for 30 seconds, 72 °C for 30 seconds) finishing with a melt curve analysis. All reactions were performed in a 7900 HT system (Applied Biosystems, Inc.). PCR primers for all sites were designed to have a melting temperature estimated at 62 °C. Reverse primers were

chosen to yield amplicons in the size range of 200-225 bp. No optimization was needed for qPCR primer sequences or for multiplex/singleplex reaction conditions.

Sanger Sequencing of 314 UAG-to-UAA loci: DNA sequencing was employed to confirm the results of the above PCR assays and to determine genotypes for 16 sites that gave ambiguous results by MASC-PCR. Amplicons 200-300 bp in length surrounding each of the 314 UAG sites were sequenced from the top-scoring clones by colony PCR as above. Sanger sequencing to confirm allelic replacements was performed by Agencourt Bioscience Corporation and the Biopolymer Facility in the Department of Genetics at Harvard Medical School. Mutations were identified by sequence alignment to the reference MG1655 genome.

Phenotype Analyses: To ensure that the codon replacements did not introduce any significant aberrant phenotypes, we conducted a number of experiments that assessed the fitness of the recoded strains. These experiments included measurements of: 1) strain growth rates, 2) auxotrophic rates and 3) frequency of recombination. Growth rate measurements were obtained by growing replicates of the recoded strains in LB^L media in 96-well plates at 30 °C and obtaining OD_{600} measurements using Molecular Devices plate readers (M5 and SpectraMax Plus). Auxotrophic rates were obtained by spotting all clonal isolates (1504) from the MAGE-cycled experiments on M9 minimal media plates (200 mL 5x M9 medium, 1 mL 1 M $MgSO_4$, 5 mL 40% glucose, 100 μL 0.5% vitamin B1 (thiamine), 1 mL D-biotin (0.25mg/mL), up to 1 L water + 15g Agar). The recombination frequency of each isolate was obtained by performing the allelic replacement protocol using a *lacZ* 90-mer oligo that produced a premature stop codon in the chromosomal *lacZ* gene. In general, 250-500 cells were plated on LB^L + Xgal/IPTG (USB Biochemicals) agar plates. Frequency of allelic replacement was calculated by dividing the

number of white colonies by the total number of colonies on plates. All phenotypic results are reported in Table S4-6.

Hierarchical Conjugation Assembly Genome Engineering (CAGE): Donor and recipient strains were grown in 3 mL LB-min containing the appropriate positive selectable antibiotics. Once cells reached logarithmic-saturated growth, 2 mL samples of each culture were transferred to 2 mL Eppendorf tubes. Cells were washed three times in order to remove antibiotics present in the growth cultures. The washing procedure consisted of centrifuging samples at 5000 rpm for 2 minutes at room temperature, removing the supernatant, and re-suspending the cell pellet in fresh LB^L containing no antibiotics. After the final wash, the donor and recipient pellets were concentrated in 100 μL LB^L in order to enhance cell-cell contact during conjugation. Conjugation was initiated by combining 80 μL of ~20x concentrated donor culture and 20 μL of ~20x concentrated recipient culture. In order to minimize RP2 pilus shearing, cells were gently mixed by pipetting. In order to minimize turbulence that can disrupt cell-cell contact during conjugation and to maximize genome transfer, the entire 100 μL donor-recipient mixture was transferred as a series of 2 x 20 μL and 6 x 10 μL spots onto an LB^L agarose plate lacking antibiotics. This conjugation plate was incubated at 32 °C for 0.5-2 hours, then the cells were re-suspended directly off of the plate using 1.5 mL LB^L and concentrated into a final volume of 250 μL. Desired recombinant genomes were selected by inoculating 5 μL of the concentrated post-conjugation culture into LB^L containing the correct combination of positive selection antibiotics (*e.g.*, 10 μg/mL zeocin, 95 μg/mL spectinomycin, and 7.5 μg/mL gentamycin). The conjugated cells that populated the selected culture were then subjected to a negative selection using either *tolC* or *galK* to ensure proper DNA transfer of UAA codons at critical junction points between donor and recipient cells (see Figure 4-4).

This engineered conjugation method was tested for the first (1/32 genome, ~143 kb) and last (1/2 genome, ~2.3 Mb) chromosomal transfer steps in the hierarchical assembly experiment (Figure 4-1). By selecting for different combinations of markers across the donor and recipient genomes and subsequent screening of specific genomic loci, recombinant clones were isolated that contained the transfer of half or full (otherwise unmodified) genomes at a frequency of ~2.5 x 10^6 (from a population of 10^9-10^{10} cells), indicating the successful DNA transfer from an integrated *oriT* with episomal expression of conjugal factors. Equivalent frequencies were observed for full genome transfers.

Upon completion of the conjugation process, we also observed the anticipated loss of the *oriT-kan* cassette in the recombinant strain. This observation yields a subtle, yet very useful feature of our engineered conjugation system. By not inheriting the *oriT* sequence, the strains are positioned to proceed to a subsequent conjugation by a one-step integration of the *oriT-kan* cassette in a new, targeted chromosomal locus (Figure 4-4A).

Illumina Whole Genome Sequencing: We prepared a paired-end Illumina sequencing library for three of the 1/8 genome strains using barcoded Illumina adapters: C21(regions 17-20), C22 (21-24), and C23 (25-28). The barcoded library was sequenced on one lane using an Illumina GAII.

Genomic DNA was prepared using a Qiagen Genome Prep kit. The purified gDNA (5 μg) was sheared to a target size of 200 bp using a Covaris E210 (estimated median band size 250bp). The sheared gDNA was PCR purified using a QIAquick PCR purification kit and end repaired (Epicentre End-it™ DNA End-Repair kit). End repair reactions consisted of the DNA sample (35 μL), 10x End repair buffer (10 μL), 1 mM dNTPs (10 μL), End repair enzyme mix (5 μL), and dH_2O (40 μL). End repair reactions were incubated at 25 °C for 30 minutes.

The end repaired DNA was PCR purified using a QIAquick PCR purification kit and A-tailed using NEB Klenow Fragment (3'→5' exo$^-$). A-tailing was performed with the DNA sample (32 μL), Klenow buffer (5 μL), 1 mM dATP (10 μL), and Klenow (3'→5' exo$^-$) (3 μL). A-tailing reactions were incubated at 37 °C for 30 minutes.

The A-tailed DNA was purified using a QIAquick PCR purification kit and Illumina PE adapters containing 3 bp barcodes (AGC for C21, CTA for C22.DO:T, and TCT for C23) were ligated. Ligations consisted of DNA sample (31 μL), 2x Rapid ligase buffer (35 μL), 50 μM Illumina PE adapters (2 μL), and Enzymatics Rapid (T4) ligase (2 μL). Ligations were incubated at 20 °C for 10 minutes, then buffer PBI was immediately added for PCR purification (QIAquick PCR purification kit).

The adapter-ligated sequencing libraries were gel purified (Qiagen Gel Purification kit) using a 2% agarose gel in 0.5x TBE (cut 2 mm bands corresponding to approximately 225 bp). The gel-purified DNA was PCR amplified using KAPA HiFi Ready Mix. The PCR reaction consisted of 2X KAPA HiFi Ready Mix (25 μL), primer PE_PCR-f (1 μL), primer PE_PCR-r (1 μL), dH2O (13 μL), and template DNA (4 μL). The PCR reaction was thermocycled as follows: 95 °C for 5 minutes; 12 cycles of 98 °C for 20 seconds, 62 °C for 15 seconds, and 72 °C for 75 seconds; 72 °C for 3 minutes. PCR products were purified using QIAquick PCR purification kit and validated by cloning using Invitrogen TOPO ZeroBlunt II according to standard protocols. The TOPO reactions were transformed into OneShot Top 10 electrocompetent cells, and a subset of colonies which were Sanger sequenced by Genewiz (M13 forward sequecing primer = GTAAAACGACGGCCAG).

The sequencing libraries were size-selected for ~225 bp bands (E-Gel® SizeSelect™ gels) and PCR purified using Qiagen MinElute columns. The libraries were quantitated by

PAGE, using a Low DNA Mass Ladder (Invitrogen), SYBR gold staining, and densitometry on a Bio-RAD geldoc.

The sequencing library was prepared by adding all 3 components (C21, C22.DO:T, and C23 sequencing libraries) to a final concentration of 10 nM. Sample QC, Clustering, and sequencing were performed by the Harvard Biopolymers Facility using Standard Illumina PE Sequencing Primers.

Genome Sequencing—Read Sorting and Processing: The raw Illumina reads in FASTQ format were preprocessed and sorted using the 3-bp barcodes in the paired end adaptors. Reads that contained anomalous barcodes were discarded. Reads containing any bases with a quality score of 2, also called the Read Segment Quality Control Indicator (based on Illumina Quality Scores by Tobia Man), were discarded at this step, but all other reads were kept. After preprocessing, all reads were exactly 34 base pairs long.

Genome Sequencing—Reference-based Assembly: The expected FASTA sequence of the EcNR2 parent strain was assembled by manually modifying the FASTA sequence of *E. coli* K-12 strain MG1655 to reflect the removal of *mutS* and the insertion of the λ prophage into the *bioAB* operon. Next, the preprocessed reads were sorted into separate files by pair group and the Burrows-Wheeler Aligner program (BWA) (*32*) was used to separately align the paired reads from each of the three strains to the expected EcNR2 FASTA sequence. The sample algorithm was used to align the reads. The distribution of insert sizes was inferred at runtime. During the read alignment step of BWA, (the aln command), a value of 10 was used for the suboptimal alignment cutoff.

Genome Sequencing—Indel and SNP Filtering: After alignment, the SAMtools package (*33*) was used to create and sort BAM files for the assemblies. From these BAM files

we generated a set of raw SNPs and short indels with respect to this reference assembly. These were then filtered using several criteria. First, using the varFilter script within SAMtools, we removed SNPs where the root mean squared mapping quality was less than 10, and indels where the root mean squared mapping quality was less than 25. We fitted the read coverage of each assembly to a gamma distribution and used the 99.95th and 0.05th percentile cutoffs for minimum and maximum read depth, beyond which SNPs and indels were discarded. We also discarded SNPs within 3 base pairs of a gap, and SNPs that occurred more densely than three within one 10 base pair window.

Genome Sequencing—Region Masking: We used custom scripts to further filter SNPs and indels by masking regions of poor assembly. We masked regions containing many truncated reads, many incorrect read pairings, many non-unique alignments, and regions with motifs known to be problematic in Illumina sequencing (GGCnG). We defined truncated read regions as those containing multiple incompletely mapped reads, separated by less than one read length, containing at least 4 truncated reads and having a number of truncated reads totaling at least one half of the length of the contiguous region in which they were found.

Regions with incorrect read pairings were defined using the following method. We found read pairs whose insert size was outside of the 99.9th and 0.1th percentile of a fitted normal distribution of mate pair distance. These reads were counted in a 34 bp rolling window. As a thresholding step we chose contiguous regions where 10 or more of these reads were found in one window length. Additionally included were contiguous regions where only one read in a pair could be mapped, and these were thresholded with a rolling window in a similar fashion, using a 6 read cutoff. As a final masking step, we removed SNPs stemming from the replacement of

amber stop codons as well as SNPs and indels where surrounding context was GGCnC, as these regions are known to be hotspots for Illumina sequencing errors.

Genome Sequencing—Annotation: After removing SNPs and indels in the masked regions as described above, we attempted to associate the remaining SNPs and indels with functional consequences. We used a modified version of Ensembl's SNP Effect Predictor software (*34*), and the Ensembl Bacteria database to find SNPs that occurred within genes. We further categorized these by synonymous and non-synonymous coding changes, frameshift mutations, premature stop mutations, mutations in the 5' and 3' UTRs, and mutations less than 100 base pairs upstream of a transcript start site (Figure S4-8). Coordinates were lifted over from ECNR2 to MG1655 to permit annotation of the SNPs and indels. This resulted in C21, C22, and C23 having 4, 5, and 5 mutations respectively having no corresponding liftover coordinates in ECNR2. These are referred to as the "unmappable" in Figure S4-8.

Supplemental material

Supplemental material for CHAPTER 4 can be found in APPENDIX C or at <http://www.sciencemag.org/content/suppl/2011/07/13/333.6040.348.DC1/Isaacs.SOM.pdf>.

References

1. A. Ambrogelly, S. Palioura, D. Soll, Natural expansion of the genetic code. *Nat Chem Biol* **3**, 29 (2007).

2. R. D. Knight, S. J. Freeland, L. F. Landweber, Rewiring the keyboard evolvability of the genetic code. *Nat. Rev. Genet.* **2**, 49 (Jan, 2001).

3. T. S. Young, P. G. Schultz, Beyond the Canonical 20 Amino Acids: Expanding the Genetic Lexicon. *J. Biol. Chem.* **285**, 11039 (Apr, 2010).

4. H. Neumann, K. Wang, L. Davis, M. Garcia-Alai, J. W. Chin, Encoding multiple unnatural amino acids via evolution of a quadruplet-decoding ribosome. *Nature* **464**, 441 (2010).

5. T. Mukai *et al.*, Codon reassignment in the Escherichia coli genetic code. *Nucleic Acids Res.* **38**, 8188 (2010).

6. P. A. Carr, G. M. Church, Genome engineering. *Nat Biotech* **27**, 1151 (2009).

7. G. Posfai *et al.*, Emergent properties of reduced-genome Escherichia coli. *Science* **312**, 1044 (May, 2006).

8. L. Y. Chan, S. Kosuri, D. Endy, Refactoring bacteriophage T7. *Mol. Syst. Biol.* **1**, 10 (2005).

9. M. Itaya, K. Tsuge, M. Koizumi, K. Fujita, Combining two genomes in one cell: Stable cloning of the Synechocystis PCC6803 genome in the Bacillus subtilis 168 genome. *Proc. Natl. Acad. Sci. U. S. A.* **102**, 15971 (Nov, 2005).

10. D. G. Gibson *et al.*, Complete chemical synthesis, assembly, and cloning of a Mycoplasma genitalium genome. *Science* **319**, 1215 (Feb, 2008).

11. C. Lartigue *et al.*, Genome transplantation in bacteria: Changing one species to another. *Science* **317**, 632 (Aug 3, 2007).

12. D. G. Gibson *et al.*, Creation of a Bacterial Cell Controlled by a Chemically Synthesized Genome. *Science* **329**, 52 (Jul, 2010).

13. J. R. Coleman *et al.*, Virus attenuation by genome-scale changes in codon pair bias. *Science* **320**, 1784 (Jun, 2008).

14. H. H. Wang *et al.*, Programming cells by multiplex genome engineering and accelerated evolution. *Nature* **460**, 894 (Aug, 2009).

15. B. Irwin, J. D. Heck, G. W. Hatfield, Codon pair utilization biases influence translational elongation step times. *J. Biol. Chem.* **270**, 22801 (Sep 29, 1995).

16. G. A. Gutman, G. W. Hatfield, Nonrandom utilization of codon pairs in *Escherichia coli*. *Proc. Natl. Acad. Sci. U. S. A.* **86**, 3699 (May, 1989).

17. H. M. Ellis, D. G. Yu, T. DiTizio, D. L. Court, High efficiency mutagenesis, repair, and engineering of chromosomal DNA using single-stranded oligonucleotides. *Proc. Natl. Acad. Sci. U. S. A.* **98**, 6742 (Jun, 2001).

18. N. Costantino, D. L. Court, Enhanced levels of lambda red-mediated recombinants in mismatch repair mutants. *Proc. Natl. Acad. Sci. U. S. A.* **100**, 15748 (Dec, 2003).

19. S. Y. Gerdes *et al.*, Experimental determination and system level analysis of essential genes in Escherichia coli MG1655. *J. Bacteriol.* **185**, 5673 (Oct, 2003).

20. R. S. Cha, H. Zarbl, P. Keohavong, W. G. Thilly, Mismatch amplification mutation assay (MAMA): application to the c-H-ras gene. *Genome Research* **2**, 14 (August 1, 1992, 1992).

21. See supporting material on Science Online.

22. P. A. Carr *et al.*, Protein-mediated error correction for de novo DNA synthesis. *Nucleic Acids Res.* **32**, 9 (2004).

23. A. B. Oppenheim, A. J. Rattray, M. Bubunenko, L. C. Thomason, D. L. Court, In vivo recombineering of bacteriophage lambda by PCR fragments and single-strand oligonucleotides. *Virology* **319**, 185 (Feb, 2004).

24. P. Funchain *et al.*, The consequences of growth of a mutator strain of Escherichia coli as measured by loss of function among multiple gene targets and loss of fitness. *Genetics* **154**, 959 (Mar, 2000).

25. H. Ochman, J. G. Lawrence, E. A. Groisman, Lateral gene transfer and the nature of bacterial innovation. *Nature* **405**, 299 (May, 2000).

26. G. R. Smith, Conjugational Recombination in Escherichia coli: Myths and Mechanisms. *Cell* **64**, 19 (Jan, 1991).

27. D. G. Yu *et al.*, An efficient recombination system for chromosome engineering in Escherichia coli. *Proc. Natl. Acad. Sci. U. S. A.* **97**, 5978 (May, 2000).

28. J. A. DeVito, Recombineering with tolC as a Selectable/Counter-selectable Marker: remodeling the rRNA Operons of Escherichia coli. *Nucleic Acids Res.* **36**, e4 (2008).

29. S. Warming, N. Costantino, D. L. Court, N. A. Jenkins, N. G. Copeland, Simple and highly efficient BAC recombineering using galK selection. *Nucleic Acids Res.* **33**, e36 (2005).

30. M. Masi, P. Vuong, M. Humbard, K. Malone, R. Misra, Initial steps of colicin E1 import across the outer membrane of Escherichia coli. *J. Bacteriol.* **189**, 2667 (Apr, 2007).

31. R. M. Schaaper, R. L. Dunn, Spectra of spontaneous mutations in Escherichia coli strains defective in mismatch correction: the nature of in vivo DNA replication errors. *Proc. Natl. Acad. Sci. U. S. A.* **84**, 6220 (1987).

32. H. Li, R. Durbin, Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* **25**, 1754 (2009).

33. H. Li *et al.*, The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**, 2078 (Aug, 2009).

34. W. McLaren *et al.*, Deriving the consequences of genomic variants with the Ensembl API and SNP Effect Predictor. *Bioinformatics* **26**, 2069 (Aug, 2010).

35. M. R. Hemm *et al.*, Small Stress Response Proteins in Escherichia coli: Proteins Missed by Classical Proteomic Studies. *J. Bacteriol.* **192**, 46 (Jan, 2010).

36. E. Jacinto-Loeza, S. Vivanco-Dominguez, G. Guarneros, J. Hernandez-Sanchez, Minigene-like inhibition of protein synthesis mediated by hungry codons near the start codon. *Nucleic Acids Res.* **36**, 4233 (Aug, 2008).

CHAPTER 5

Genomically Recoded Organisms Impart New Biological Functions

This chapter is reproduced with minor edits with permission from its initial publication:

Lajoie MJ, Rovner AJ, Goodman DB, Aerni HR, Haimovich AD, Kuznetsov G, Mercer JA, Wang HH, Carr PA, Mosberg JA, Rohland N, Schultz PG, Jacobson JM, Rinehart J, Church GM, Isaacs FI (2013) *Genomically Recoded Organisms Impart New Biological Functions*. **Science** 342: 357-60.

Research contributions: M. Lajoie, H. Wang, P. Carr, J. Jacobson, G. Church, and F. Isaacs conceived of the project. M. Lajoie, A. Rovner, and F. Isaacs completed construction of strain C321. M. Lajoie, D. Goodman, G. Kuznetsov, J. Mercer, N. Rohland, and F. Isaacs performed the genome sequencing and fitness analysis. M. Lajoie, A. Rovner, and F. Isaacs performed the NSAA assays. H. Aerni, J. Rinehart, and F. Isaacs performed mass spectrometry analysis. M. Lajoie performed all phage assays. M. Lajoie, J. Rinehart, G. Church, and F. Isaacs planned the experiments. M. Lajoie wrote the paper. All authors reviewed and edited the manuscript. M. Lajoie, G. Church, and F. Isaacs oversaw all aspects of the project.

Acknowledgements: We dedicate this paper to the memory of our friend, colleague and gifted scientist, Tara Gianoulis. We thank Roberto Kolter for JC411, David Reich for help with sequencing libraries, Christine and Jonathan Seidman for Covaris E210; John Aach, Sri Kosuri, and Uri Laserson for bioinformatics; Travis Young, Francis Peters, and Wayne Barnes for NSAA incorporation advice; Ian Molineux and Sri Kosuri for phage advice; Sara Vassallo and Prashant Mali for experimental support; Dieter Söll, Tony Forster, Ting Wu, Ken Oye, Chris Gregg, Michael Napolitano, Uri Laserson, Adrian Briggs, Dan Mandell, and Raj Chari for helpful comments. Funding was from Department of Energy [DE-FG02-02ER63445], NSF [SA5283-11210], NIH [NIDDK-K01DK089006 to J.R.], DARPA [N66001-12-C-4040, N66001-12-C-4020, N66001-12-C-4211], Arnold and Mabel Beckman Foundation (F.J.I.), Department of Defense NDSEG Fellowship (M.J.L.), NIH Director's Early Independence Award (1DP5OD009172-01 to H.H.W), and the Assistant Secretary of Defense for Research and Engineering (Air Force Contract #FA8721-05-C-0002 to P.A.C.). Opinions, interpretations, conclusions and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

Abstract

We describe the construction and characterization of a genomically recoded organism (GRO). We replaced all known UAG stop codons in *Escherichia coli* MG1655 with synonymous UAA codons, which permitted the deletion of release factor 1 and reassignment of UAG translation function. This GRO exhibited improved properties for incorporation of nonstandard amino acids that expand the chemical diversity of proteins *in vivo*. The GRO also exhibited increased resistance to T7 bacteriophage, demonstrating that new genetic codes could enable increased viral resistance.

Introduction

The conservation of the genetic code permits organisms to share beneficial traits through horizontal gene transfer (*1*), and enables the accurate expression of heterologous genes in nonnative organisms (*2*). However, the common genetic code also allows viruses to hijack host translation machinery (*3*) and compromise cell viability. Additionally, genetically modified organisms (GMOs) can release functional DNA into the environment (*4*). Virus resistance (*5*) and biosafety (*6*) are among today's major unsolved problems in biotechnology, and no general strategy exists to create genetically isolated or virus-resistant organisms. Furthermore, biotechnology has been limited by the 20 amino acids of the canonical genetic code, which use all 64 possible triplet codons, limiting efforts to expand the chemical properties of proteins by means of nonstandard amino acids (NSAAs) (*7, 8*).

Changing the genetic code could solve these challenges and reveal new principles that explain how genetic information is conserved, encoded, and exchanged (Figure S5-1). We propose that genomically recoded organisms (GROs, whose codons have been reassigned to

create an alternate genetic code) would be genetically isolated from natural organisms and viruses, as horizontally transferred genes would be mistranslated, producing nonfunctional proteins. Furthermore, GROs could provide dedicated codons to improve the purity and yield of NSAA-containing proteins, enabling robust and sustained incorporation of more than 20 amino acids as part of the genetic code.

Results

We constructed a GRO in which all instances of the UAG codon have been removed, permitting the deletion of release factor 1 (RF1; terminates translation at UAG and UAA) and, hence, eliminating translational termination at UAG codons. This GRO allows us to reintroduce UAG codons, along with orthogonal translation machinery [*i.e.*, aminoacyl–tRNA synthetases (aaRSs) and tRNAs] (*7, 9*), to permit efficient and site-specific incorporation of NSAAs into proteins (Figure 5-1). That is, UAG has been transformed from a nonsense codon (terminates translation) to a sense codon (incorporates amino acid of choice), provided the appropriate translation machinery is present. We selected UAG as our first target for genome-wide codon reassignment because UAG is the rarest codon in *Escherichia coli* MG1655 (321 known instances), prior studies (*7, 10*) demonstrated the feasibility of amino acid incorporation at UAG, and a rich collection of translation machinery capable of incorporating NSAAs has been developed for UAG (*7*).

We used an *in vivo* genome editing approach (*11*), which is more efficient than *de novo* genome synthesis at exploring new genotypic landscapes and overcoming genome design flaws. Although a single lethal mutation can prevent transplantation of a synthetic genome (*12*), our approach allowed us to harness genetic diversity and evolution to overcome any potential

deleterious mutations at a cost considerably less than *de novo* genome synthesis (supplemental text "Time and Cost" section). In prior work, we used multiplex automated genome engineering [MAGE (*13*)] to remove all known UAG codons in groups of 10 across 32 *E. coli* strains (*11*), and conjugative assembly genome engineering [CAGE (*11*)] to consolidate these codon changes in groups of ~80 across four strains. In this work, we overcome technical hurdles (supplemental text) to complete the assembly of the GRO and describe the biological properties derived from its altered genetic code.

The GRO [C321.ΔA, named for 321 UAG→UAA conversions and deletion of *prfA* (encodes RF1, Table 5-1)] and its $RF1^{+}$ precursor (C321) exhibit normal prototrophy and morphology (Figure S5-2) with 60% increased doubling time compared with *E. coli* MG1655 (Table S5-1). Genome sequencing (GenBank accession CP006698) confirmed that all 321 known UAGs were removed from its genome and that 355 additional mutations were acquired during construction (10^{-8} mutations per base pair per doubling over ~7340 doublings; Figure S5-3 and Tables S5-2 to S5-4).

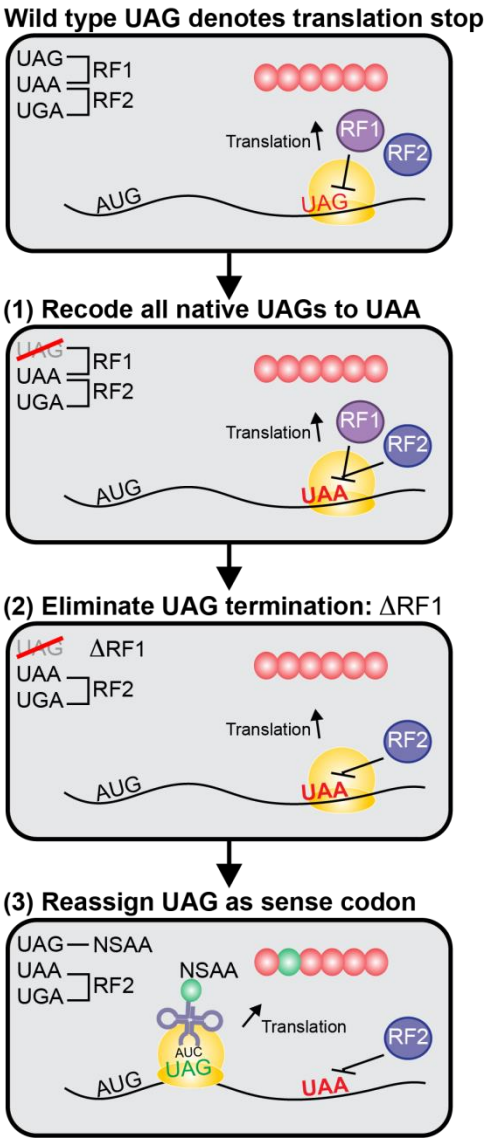


Figure 5-1. Engineering a GRO with a reassigned UAG codon. Wild-type *E. coli* MG1655 has 321 known UAG codons that are decoded as translation stops by RF1 (for UAG and UAA). (1) Remove codons: converted all known UAG codons to UAA, relieving dependence on RF1 for termination. (2) Eliminate natural codon function: abolished UAG translational termination by deleting RF1, creating a blank codon. (3) Expand the genetic code: introduced an orthogonal aminoacyl-tRNA synthetase (aaRS) and tRNA to reassign UAG as a dedicated sense codon capable of incorporating nonstandard amino acids (NSAAs) with new chemical properties.

Although maintaining the *E. coli* MG1655 genotype was not a primary goal of this work, future applications requiring increased genome stability could exploit reversible switching of *mutS* function (*14*) to reduce off-target mutagenesis. CAGE improved the fitness of several strains in the C321 lineage (Figure S5-3), implicating off-target mutations in the reduced fitness.

C321.ΔA exhibited improved performance compared with previous strategies for UAG codon reassignment (*15, 16*), permitting the complete reassignment of UAG from a stop codon to a sense codon capable of incorporating NSAAs into proteins. One previous strategy used a variant of release factor 2 (RF2) that exhibits enhanced UAA termination (*16*) and weak UAG termination (*17*). The second strategy substituted a UAA stop codon in each of the seven essential genes naturally terminating with UAG (Table S5-5) and reduced ribosome toxicity by efficiently incorporating amino acids at the remaining 314 UAGs (*15*). For comparative purposes, we used MAGE to create strains C0.B*.ΔA::S [expresses enhanced RF2 variant (*16*)], C7.ΔA::S (UAG changed to UAA in seven essential genes), and C13.ΔA::S [UAG changed to UAA in seven essential genes plus six nonessential genes (Table S5-5)] (Table 5-1). C refers to the number of codon changes, while A and B refer to *prfA* (RF1) and *prfB* (RF2) manipulations, respectively. In contrast to previous work (*15*), we deleted RF1 in these strains without introducing a UAG suppressor, perhaps because near-cognate suppression is increased in *E. coli* MG1655 (*18*). Nevertheless, these strains exhibited a strong selective pressure to acquire UAG suppressor mutations (see below).

To assess the fitness effects of RF1 removal and UAG reassignment, we measured the doubling time and maximum cell density of each strain (Table S5-1 and Figure S5-4). We found that C321 was the only strain for which RF1 removal and UAG reassignment was not deleterious (Figure 5-2). Because we did not modify RF2 to enhance UAA termination (*16*), this confirms

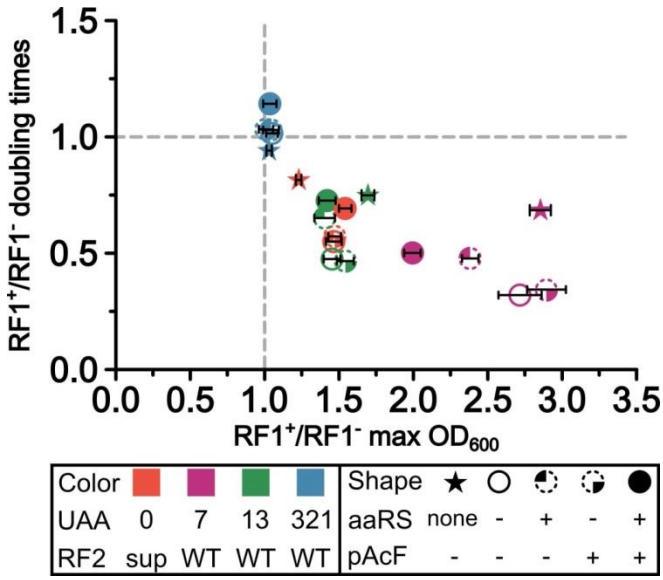


Figure 5-2. Effects of UAG reassignment at natural UAG codons. Ratios of maximum cell densities (horizontal axis) and doubling times (vertical axis) were determined for $RF1^+$ strains versus their corresponding $RF1^-$ strains (n = 3) in the presence or absence of UAG suppression. Symbol color specifies genotype: **UAA** is the number of UAG→UAA mutations, and **RF2** is "WT" (wild-type) or "sup" [RF2 variant that can compensate for RF1 deletion (*16*)]. Symbol shape specifies NSAA expression: **aaRS** (aminoacyl-tRNA synthetase) is "none" (genes for UAG reassignment were absent), "-" [pEVOL-pAcF (*9*) is present but not induced, so only the constitutive aaRS and tRNA are expressed], or "+" (pEVOL-pAcF is fully induced using L-arabinose), and **pAcF** is "-" (excluded) or "+" (supplemented). Strains that do not rely on RF1 are expected to have a $RF1^+/RF1^-$ ratio at (1,1). $RF1^-$ strains exhibiting slower growth are below the horizontal gray line, and $RF1^-$ strains exhibiting lower maximum cell density are to the right of the vertical gray line. The doubling time error bars are too small to visualize.

that RF1 is essential only for UAG translational termination and not for UAA termination or other essential cellular functions. By contrast, RF1 removal significantly impaired fitness for C0.B*.ΔA::S, and codon reassignment exacerbated this effect (Figure 5-2 and Figure S5-5), probably because NSAA incorporation outcompeted the weak UAG termination activity (*17*) exerted by the RF2 variant (*16*). C7.ΔA::S and C13.ΔA::S also exhibited strongly impaired fitness, likely due to more than 300 non-essential UAG codons stalling translation in the absence of RF1-mediated translation at UAG codons (*15*); accordingly, *p*-acetylphenylalanine (pAcF) incorporation partially alleviated this effect (Figure 5-2). However, not all NSAAs improved fitness in partially recoded strains; phosphoserine (Sep) impairs fitness in similar strains (*19*), perhaps by causing proteome-scale misfolding. Together, these results indicate that only the complete removal of all instances of the UAG codon overcomes these deleterious effects; therefore, it may be the only scalable strategy for sustained NSAA translation and for complete reassignment of additional codons.

We tested the capacity of our recoded strains to efficiently incorporate NSAAs [pAcF, *p*-azidophenylalanine (pAzF), or 2-naphthalalanine (NapA)] into Green Fluorescent Protein (GFP)

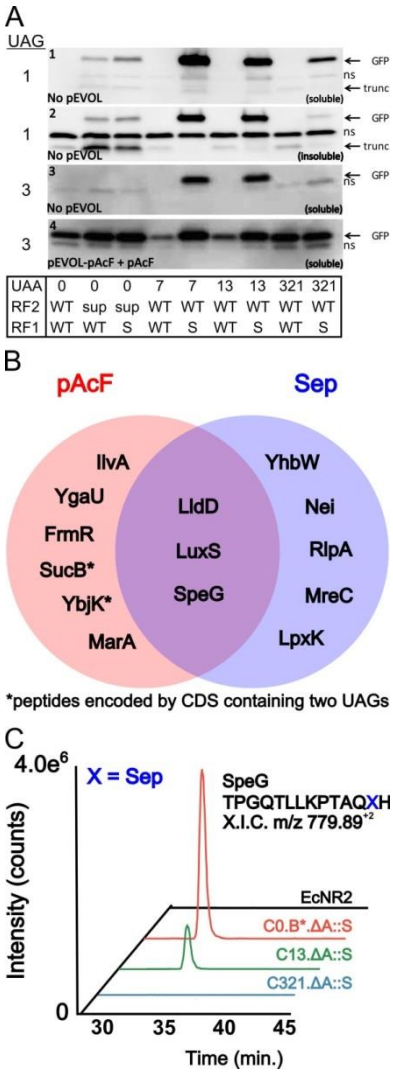


Figure 5-3. NSAA incorporation in GROs. **(A)** Western blots demonstrate that C0.B*.ΔA::S terminates at UAG in the absence of RF1 and that C7.ΔA::S and C13.ΔA::S have acquired natural suppressors that allow strong NSAA-independent read-through of three UAG codons. When pAcF was omitted, one UAG reduced the production of full-length GFP, and three UAGs reduced production to undetectable levels for all strains except for C7.ΔA::S and C13.ΔA::S, demonstrating that undesired near-cognate suppression (*18*) is weak for most strains even when RF1 is inactivated. However, all strains show efficient translation through three UAG codons when pAcF is incorporated. Western blots were probed with an antibody to GFP that recognizes an N-terminal epitope. **UAA** is the number of UAG→UAA mutations; **RF2** is "WT" (wild-type) or "sup" [RF2 variant that can compensate for RF1 deletion (*16*)]; **RF1** is "WT" (wild-type) or "S" (Δ*prfA*::*spec*^{R}). "GFP" is full-length GFP; "trunc" is truncated GFP from UAG termination and is enriched in the insoluble fraction; "ns" indicates a non-specific band. **(B)** Venn diagram representing NSAA-containing peptides detected by mass spectrometry in C0.B*.ΔA::S when UAG was reassigned to incorporate *p*-acetylphenylalanine (pAcF, red) or phosphoserine (Sep, blue). No NSAA-containing peptides were identified in C321.ΔA::S. Asterisk (*) indicates coding DNA sequence possessing two tandem UAG codons. **(C)** Extracted ion chromatograms are shown for UAG suppression of the SpeG peptide to investigate Sep incorporation in natural proteins. Peptides containing Sep were only observed in C0.B*.ΔA::S, C7.ΔA::S, and C13.ΔA::S, as Sep incorporation was below the detection limit in EcNR2 (RF1^{+}), and *speG* was recoded in C321.ΔA::S.

variants containing zero, one, or three UAG codons (Figure 5-3 and Figure S5-6). In the presence of NSAAs, the RF1^{+} strains efficiently read through variants containing three UAGs, demonstrating that the episomal pEVOL translation system, which expresses an aaRS and tRNA that incorporate a NSAA at UAG codons (*9*), is extremely active and strongly outcompetes RF1. In the absence of NSAAs, the RF1^{-} strains exhibited detectable amounts of near-cognate suppression (*18*) of a single UAG. C321.ΔA::S exhibited strong expression of UAG-containing GFP variants only in the presence of the correct NSAA, whereas C7.ΔA::S and C13.ΔA::S displayed read-through of all three UAG codons even in the absence of NSAAs, suggesting efficient incorporation of natural amino acids at native UAGs (*17*). Mass spectrometry indicated that C13.ΔA::S incorporated Gln, Lys, and Tyr at UAG codons. DNA sequencing in C7.ΔA::S and C13.ΔA::S revealed UAG suppressor mutations in *glnV*, providing direct genetic evidence of Gln suppression observed by Western blot (Figure 5-

3A) and mass spectrometry (Table S5-13). C0.B*.ΔA::S displayed truncated GFP variants corresponding with UAG termination in the absence of RF1 (*17*) (Figure 5-3A).

We directly investigated the impact of pAcF and Sep incorporation on the proteomes (Figure 5-3B) (*20*) of our panel of strains (Table 5-1) using mass spectrometry (Tables S5-6 to S5-12). No Sep-containing peptides were observed for EcNR2, illustrating that RF1 removal is necessary for NSAA incorporation by the episomal phosphoserine system (*21*), which is an inefficient orthogonal translation machinery (*19*) (Figure 5-3C and Table S5-10). By contrast, we observed NSAA-containing peptides in unrecoded (C0.B*.ΔA::S) and partially recoded (C13.ΔA::S) strains, and not the GRO (C321.ΔA::S), which lacks UAGs in its genome (Figure 5-3, B and C, Figure S5-7, and Tables S5-6 to S5-12). Such undesired incorporation of NSAAs (or natural amino acids) likely underlies the fitness impairments observed for C0.B*.ΔA::S, C7.ΔA::S, and C13.ΔA::S. In contrast to the other $RF1^-$ strains, C321.ΔA::S demonstrated equivalent fitness to its $RF1^+$ precursor (Figure 5-2), and efficiently expressed all GFP variants without incorporating NSAAs at unintended sites (Figures 5-2, 5-3, and S5-6). Therefore, complete UAG removal is the only strategy that provides a devoted codon for plug-and-play NSAA incorporation without impairing fitness (Figures 5-2 and 5-3).

To determine whether this GRO can obstruct viral infection, we challenged $RF1^-$ strains with bacteriophages T4 and T7. Viruses rely on their host to express proteins necessary for propagation. Because hosts with altered genetic codes would mistranslate viral proteins (*3*), recoding may provide a general mechanism for resistance to all natural viruses. Given that UAG codons occur rarely and only at the end of genes, we did not expect UAG reassignment to result in broad phage resistance. Although the absence of RF1 did not appear to affect T4 (19 of 277

stop codons are UAG), it significantly enhanced resistance to T7 (6 of 60 stop codons are UAG) (Figure 5-4).

Table 5-1. Recoded strains and their genotypes

Strain[a]	**Essential codons changed**[b]	**Total codons changed**[c]	**Previously essential codon functions manipulated**[d]	**Expected (obs.) UAG translation function**[e]
EcNR2	0	0	None	Stop
C0.B*	0	0	*prfB*‡	Stop
C0.B*.ΔA::S	0	0	*prfB*‡, Δ*prfA*::*spec*^{R}	None (stop*)
C7	7	7	None	Stop
C7.ΔA::S	7	7	Δ*prfA*::*spec*^{R}	None (sup)
C13	7	13	None	Stop
C13.ΔA::S	7	13	Δ*prfA*::*spec*^{R}	None (sup)
C321	7	321	None	Stop
C321.ΔA::S	7	321	Δ*prfA*::*spec*^{R}	None (nc)
C321.ΔA::T	7	321	Δ*prfA*::*tolC*	None (nc)
C321.ΔA	7	321	Δ*prfA*	None (nc)

[a]All strains are based on EcNR2 {*E. coli* MG1655 Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*] Δ*mutS*::*cat*} which is mismatch repair deficient (Δ*mutS*) to achieve high frequency allelic replacement; C0 and C321 strains are Δ*mutS*::*zeo*^{R}; C7 and C13 strains are Δ*mutS*::*tolC*; C7, C13, and C321 strains have the endogenous *tolC* deleted, making it available for use as a selectable marker. Spectinomycin resistance (S) or tolC (T) were used to delete *prfA* (A). Bacterial genetic nomenclature describing these strains includes :: (insertion) and Δ (deletion).

[b]Out of a total of 7

[c]Out of a total of 321

[d]*prfA* encodes RF1, terminating UAG and UAA; *prfB* encodes RF2, terminating UGA and UAA; *prfB*‡ = RF2 variant (T246A, A293E, and removed frameshift) exhibiting enhanced UAA termination (*16*) and weak UAG termination (*17*).

[e]Observed translation function: Stop = expected UAG termination; stop* = weak UAG termination from RF2 variant; sup = strong selection for UAG suppressor mutations; nc = weak near-cognate suppression (*i.e.*, reduced expression compared to C7.ΔA::S and C13.ΔA::S) in the absence of all other UAG translation function.

RF1$^-$ hosts produced significantly smaller T7 plaques independent of host doubling time (Figures 5-4A and S5-8). The only exception was C0.B*.ΔA::S, which produced statistically equivalent plaque sizes regardless of whether RF1 was present (Figure 5-4A, Table S5-14). Consistent with the observation that the modified RF2 variant could weakly terminate UAG [(*17*) and herein], our results suggest that C0.B*.ΔA::S terminates UAG codons well enough to support normal T7 infection.

Given that plaque area and phage fitness (doublings per hour) do not always correlate, we confirmed that T7 infection is inhibited in RF1$^-$ hosts by comparing T7 fitness and lysis time in C321 versus C321.ΔA (Figure 5-4B). Phage fitness (doublings per hour) is perhaps the most

relevant measure for assessing phage resistance because it indicates how quickly a log-phase phage infection expands (*22*). We found that T7 fitness was significantly impaired in strains lacking RF1 (P = 0.002), and kinetic lysis curves (Figure S5-9) confirmed that lysis was significantly delayed in the absence of RF1 (P < 0.0001, Figure 5-4B). Meanwhile, one-step growth curves (Figure S5-10) indicated that burst size (average number of phages produced per lysed cell) in RF1$^-$ hosts was also reduced by 59% (± 9%), and phage packaging was delayed by 30% (± 2%) (Table S5-15). We hypothesize that ribosome stalling at the gene 6 (T7 exonuclease) UAG explains the T7 fitness defect in RF1$^-$ hosts, whereas T4 may not possess a UAG-terminating essential gene with a similar sensitivity (supplemental text). Abolishing the function of additional codons could block the translation of viral proteins and prevent infections entirely.

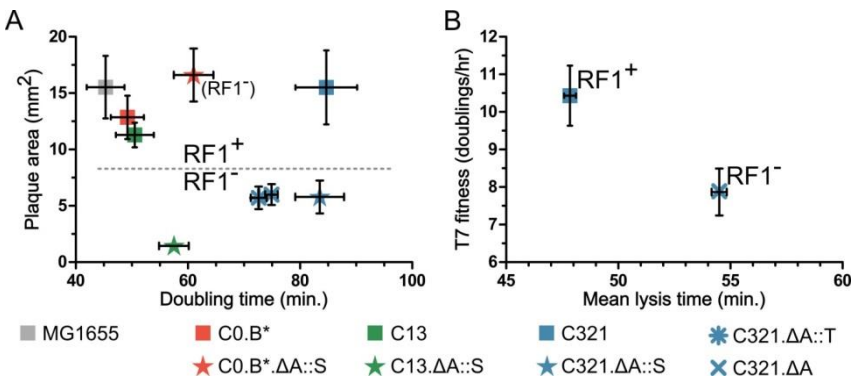


Figure 5-4. Bacteriophage T7 infection is attenuated in GROs lacking RF1. RF1 (*prfA*) status is denoted by symbol shape: ■ is wt *prfA* (WT), ★ is Δ*prfA*::*spec*^{R} (ΔA::S), ✱ is Δ*prfA*::*tolC* (ΔA::T), and ✕ is a clean deletion of *prfA* (ΔA). **(A)** RF1 status affects plaque area (Kruskal-Wallis one-way analysis of variance, P < 0.001), but strain doubling time does not (Pearson correlation, P = 0.49). Plaque areas (mm^2) were calculated with ImageJ, and means ± 95% confidence intervals are reported (n > 12 for each strain). In the absence of RF1, all strains except for C0.B*.ΔA::S yielded significantly smaller plaques, indicating that the RF2 variant (*16*) can terminate UAG adequately to maintain T7 fitness. A statistical summary can be found in Table S5-14. **(B)** T7 fitness (doublings/hr) (*22*) is impaired (P = 0.002) and mean lysis time (min) is increased (P < 0.0001) in C321.ΔA compared to C321. Significance was assessed for each metric by using an unpaired t test with Welch's correction.

Discussion

Using multiplex genome editing, we removed all instances of the UAG codon and reassigned its function in the genome of a living cell. The resulting GRO possesses a devoted

UAG sense codon for robust NSAA incorporation that is suitable for industrial protein production. GROs also establish the basis for genetic isolation and virus resistance, and additional recoding will help fully realize these goals—additional triplets could be reassigned, unnatural nucleotides could be used to produce new codons (*23*), and individual triplet codons could be split into several unique quadruplets (*8, 24*) that each encode their own NSAA. In an accompanying study (*25*), we show that twelve additional triplet codons may be amenable to removal and eventual reassignment in *E. coli*. However, codon usage rules are not fully understood, and recoded genome designs are likely to contain unknown lethal elements. Thus, it will be necessary to sample vast genetic landscapes, efficiently assess phenotypes arising from individual changes and their combinations, and rapidly iterate designs to change the genetic code at the genome level.

Materials and Methods

All DNA oligonucleotides were purchased with standard purification and desalting from Integrated DNA Technologies (Table S5-19). Unless otherwise stated, all cultures were grown in LB-Lennox medium (LB^L, 10 g/L bacto tryptone, 5 g/L sodium chloride, 5 g/L yeast extract) with pH adjusted to 7.45 using 10 M NaOH. LB^L agar plates were LB^L plus 15 g/L bacto agar. Top agar was LB^L plus 7.5 g/L bacto agar. MacConkey agar was prepared using BD Difco™ MacConkey agar base according to the manufacturer's protocols. M9 medium (6 g/L Na_2HPO_4, 3 g/L KH_2PO_4, 1 g/L NH_4Cl, 0.5 g/L NaCl, 3 mg/L $CaCl_2$) and M63 medium (2 g/L $(NH_4)_2SO_4$, 13.6 g KH_2PO_4, 0.5 mg $FeSO_4 \cdot 7H_2O$) were adjusted to pH 7 with 10 M NaOH and KOH, respectively. Both minimal media were supplemented with 1 mM $MgSO_4 \cdot 7H_2O$, 0.083 nM thiamine, 0.25 µg/L D-biotin, and 0.2% w/v carbon source (galactose, glycerol, or glucose).

The following selective agents were used: carbenicillin (50 µg/mL), chloramphenicol (20 µg /mL), kanamycin (30 µg/mL), spectinomycin (95 µg/mL), tetracycline (12 µg/mL), zeocin (10 µg/mL), gentamycin (5 µg/mL), SDS (0.005% w/v), Colicin E1 (ColE1; ~10 µg/mL), and 2-deoxygalactose (2-DOG; 0.2%). ColE1 was expressed in strain JC411 and purified as previously described (*26*). All other selective agents were obtained commercially.

The following inducers were used at the specified concentrations unless otherwise indicated: anhydrotetracycline (30 ng/µL), L-arabinose (0.2% w/v). *p*-acetyl-L-phenylalanine (pAcF) was purchased from PepTech (# AL624-2) and used at a final concentration of 1 mM. O-phospho-L-serine (Sep) was purchased from Sigma Aldrich (# P0878-25G) and used at a final concentration of 2 mM.

Strains: All strains were based on EcNR2 (*11*) (*Escherichia coli* MG1655 Δ*mutS*::*cat* Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*]). Strains C321 [strain 48999 (www.addgene.org/48999)] and C321.ΔA [strain 48998 (www.addgene.org/48998)] are available from addgene.

Selectable marker preparation: Selectable markers were prepared using primers described in Table S5-19. PCR reactions (50 µL per reaction) were performed using Kapa HiFi HotStart ReadyMix according to the manufacturer's protocols with annealing at 62 °C. PCR products were purified using the Qiagen PCR purification kit, eluted in 30 µL of dH_2O, quantitated using a NanoDrop™ ND1000 spectrophotometer, and analyzed on a 1% agarose gel with ethidium bromide staining to confirm that the expected band was present and pure.

MAGE and λ Red-mediated recombination: MAGE (*13*), CoS-MAGE (*14*), and λ Red-mediated recombination (*27*) were performed as previously described. Briefly, an overnight culture was diluted 100-fold into 3 mL LB^L plus antibiotics and grown at 30 °C in a rotator drum

until mid-log growth was achieved (OD_{600} ~0.4-0.6). Lambda Red was induced in a shaking water bath (42 °C, 300 rpm, 15 minutes), then induced culture tubes were cooled rapidly in an ice slurry for at least two minutes. Electrocompetent cells were prepared at 4 °C by pelleting 1 mL of culture (centrifuge at 16,000 rcf for 20 seconds) and washing the cell pellet twice with 1 mL ice cold deionized water (dH_2O). Electrocompetent pellets were resuspended in 50 µL of dH_2O containing the desired DNA. For MAGE oligos, no more than 5 µM (0.5 µM of each oligo) was used. For CoS-MAGE, no more than 5.5 µM (0.5 µM of each oligo including the co-selection oligo) was used. For dsDNA PCR products, 50 ng was used. Cells were transferred to 0.1 cm cuvettes, electroporated (BioRad GenePulser™, 1.78 kV, 200 Ω, 25 µF), and then immediately resuspended in 3 mL LB^L (MAGE and CoS-MAGE) or 1.5 mL LB^L (dsDNA). Recovery cultures were grown at 30 °C in a rotator drum. For continued MAGE cycling, cultures were recovered to mid-log phase before being induced for the next cycle. To isolate monoclonal colonies, cultures were recovered for at least 3 hours (MAGE and CoS-MAGE) or 1 hour (dsDNA) before plating on selective media. For *tolC* and *galK* negative selections, cultures were recovered for at least 7 hours to allow complete protein turnover before exposure to ColE1 and 2-deoxygalactose, respectively.

CAGE: CAGE was performed as previously described (*11*). Briefly, conjugants were grown to late-log phase in all relevant antibiotics (including tetracycline in the donor culture to select for the presence of conjugal plasmid pRK24 (*28*)). At mid-log growth, 2 mL of each culture was transferred to a 2 mL microcentrifuge tube and pelleted (5000 rcf, 5 minutes). Cultures were washed twice with LB^L to remove antibiotics, then the pellets were resuspended in 100 µL LB^L. Donor (10 µL) and recipient (90 µL) samples were mixed by gentle pipetting and then spotted onto a pre-warmed LB^L agar plate (6 x 10 µL and 2 x 20 µL spots). Conjugation

proceeded at 30 °C without agitation for 1 – 24 hours. Conjugated cells were resuspended off of the LB^L agar plate using 750 µL liquid LB^L, and then 3 µL of the resuspended conjugation was inoculated into 3 mL of liquid LB^L containing the appropriate selective agents. The population with the correct resistance phenotype was then subjected to ColE1 negative selection to eliminate cells that retained *tolC*.

Each round of conjugation, genotyping, and strain manipulation required a minimum of 5 days to complete. On day 1, the conjugation and positive selections were performed. On day 2, the population of cells exhibiting the desired resistance phenotype was subjected to a ColE1 selection to eliminate candidates that retained *tolC*. The ColE1-resistant population was then spread onto plates to isolate monoclonal colonies. On day 3, candidate colonies were grown in a 96-well format and screened for the desired genotypes via PCR (to confirm loss of *tolC*) and MASC-PCR (to confirm the presence of the desired codon replacements). On day 4, *tolC* or *kanR*-oriT was recombined directly into one of the positive markers, and recombinants were plated on LB^L plates containing SDS or kanamycin, respectively. On day 5, candidate colonies were grown in liquid LB^L containing SDS or kanamycin and used as PCR template to confirm successful replacement of positive selection markers with *tolC* or *kanR*-oriT. These strains were ready for the next conjugation.

Positive/Negative selections:

Positive selection for tolC: TolC provides robust resistance to SDS (0.005% w/v) in LB^L (both liquid and LB^L agar).

Negative selection for tolC: After *tolC* was removed via λ Red-mediated recombination or conjugation, cultures were recovered for at least 7 hours prior to ColE1 selection. This was enough time for the recombination to proceed and for complete protein turnover in the

recombinants (*i.e.* residual TolC protein no longer present). ColE1 selections were performed as previously described (*11*). Briefly, pre-selection cultures were grown to mid-log phase (OD_{600} ~0.4), then diluted 100-fold into 150 µL of LB^L and LB^L + ColE1. Once growth was detected, monoclonal colonies were isolated on non-selective plates and PCR screened to confirm the loss of *tolC*.

Positive selection for galK: GalK is necessary for growth on galactose (0.2% w/v) as a sole carbon source. It is important to thoroughly wash the cells with M9 media to remove residual carbon sources prior to selection in M63 + galactose (both liquid and M63 agar). Noble agar must be used, since Bacto agar may contain contaminants that can be used as alternative carbon sources.

Negative selection for galK: After *galK* was removed via λ Red-mediated recombination or conjugation, cultures were recovered for at least 7 hours prior to 2-DOG selection. This was enough time for the recombination to proceed and for complete protein turnover in the recombinants (*i.e.* residual GalK protein no longer present). 2-DOG selections were performed as previously described (*29*). Briefly, pre-selection cultures were grown to mid-log phase (OD_{600} ~0.4), washed three times in M9 medium to remove residual nutrients from LB^L, and then inoculated into M63 + 0.2% glycerol and M63 + 0.2% glycerol + 0.2% 2-DOG. Once growth was detected, monoclonal colonies were isolated on non-selective plates (LB^L agar or MacConkey agar) and PCR screened to confirm the loss of *galK*. When possible, colonies were streaked onto MacConkey + 0.2% galactose indicator plates (white colonies are Gal- and red colonies are Gal+) prior to PCR screening, but MacConkey media is toxic to strains that do not express TolC, which provides resistance to bile salts. We also found that 2-DOG selection was

effective in LB^L, but PCR screening was important because LB^L + 2-DOG selection was less stringent.

Screening for galK and malK: Cultures were diluted and plated for single colonies on MacConkey agar + 0.2% galactose (galK) or MacConkey agar + 0.2% maltose (malK) indicator plates (white colonies are Gal- or Mal-, and red colonies are Gal+ or Mal+). The genotypes were confirmed *via* PCR.

Genotyping: After λ Red-mediated recombination or conjugation, colony PCR was used to confirm the presence or absence of selectable markers at desired positions. Colony PCR (10 µL per reaction) was performed using Kapa 2G Fast HotStart ReadyMix according to the manufacturer's protocols with annealing at 56 °C. Results were analyzed on a 1% agarose gel with ethidium bromide staining.

Multiplex allele-specific colony PCR (MASC-PCR) was used to simultaneously detect up to 10 UAG→UAA conversions as previously described (*11*). Briefly, each allele was interrogated by two separate PCRs to detect the UAG/UAA status. The two reactions shared the same reverse primer but used different forward primers whose 3′ ends annealed to the SNP being assayed. Amplification only in the wt-detecting PCR indicated a UAG allele, whereas amplification only in the mutant-detecting PCR indicated a UAA allele. Each primer set produced a unique amplicon size corresponding to its target allele (100, 150, 200, 250, 300, 400, 500, 600, 700 and 850 bp). Template was prepared by growing monoclonal colonies to late-log phase in 150 µl LB^L and then diluting 2 µl of culture into 100 µl dH_2O. Initially, we used Qiagen Multiplex PCR kit, but KAPA 2G Fast Multiplex Ready Mix produced cleaner, more even amplification across our target amplicon size ranges. Therefore, typical MASC-PCR reactions contained KAPA 2G Fast Multiplex ReadyMix (Kapa Biosystems, # KK5802) and 10X Kapa

dye in a final volume of 10 µl, including 2 µl of template and 0.2 µM of each primer. PCR activation occurred at 95°C (3 min), followed by 27 cycles of 95°C (15 sec), 63–67°C (30 sec; annealing temperature was optimized for each set of MASC-PCR primers), and 72°C (70 sec). The final extension was at 72°C (5 min). MASC-PCR results were analyzed on 1.5% agarose gels with ethidium bromide staining to ensure adequate band resolution.

Sanger sequencing was performed by Genewiz or Eton Bioscience, Inc.

Genomic DNA for whole genome sequencing was prepared using a Qiagen Genomic DNA purification kit or by simultaneously lysing raw culture and shearing genomic DNA using a Covaris E210 AFA Ultrasonication machine. Illumina libraries were prepared as previously described (*30*). Each strain was barcoded with a unique 6 bp barcode for pooling. Up to 16 strains were pooled for sequencing on a single HiSeq lane, and up to 4 genomes were pooled for sequencing on a single MiSeq lane. Whole genome sequencing was performed using Illumina HiSeq or MiSeq systems. The HiSeq samples were sequenced with paired end 50 bp or 100 bp reads, and the MiSeq samples were sequenced with paired end 150 bp reads.

Sequencing analysis: In order to analyze the sequencing data from 68 distinct genomes, we developed a software pipeline that connects several modular tools and custom scripts for analysis and visualization. The goal of our pipeline was to identify SNPs and structural variants relative to the reference genome *E. coli* K-12 MG1655 (U00096.2, GI:48994873). Note that we use the term SNP to mean any small mismatches or indels identified by Freebayes (<22 bp). We use the term structural variant to refer to large insertions detected by Breakdancer and Pindel, deletions, or other significant junction events (confirmed variants of size 170 bp and 776 bp in C321.ΔA).

FASTQ conversion to SAM/BAM: FASTQ reads were split using individual genome barcodes with the FASTX toolkit (*31*). After splitting and trimming of the 6 bp barcode, FASTQ files for individual reads were aligned to the reference genome (*E. coli* K-12 MG1655 or the C321.ΔA predicted genome sequence) using Bowtie2 version 2.0.0-beta5 (*32*) with local alignment and soft-clipping enabled. PCR duplicates were removed using the Picard toolkit <http://picard.sourceforge.net/> and reads were realigned around short indels using the Genome Analysis Toolkit (*33*).

SNP calling using Freebayes: SNPs were called using the Freebayes package (arXiv:1207.3907v2 [q-bio.GN]). SNP calls were made using a *--ploidy* flag value of 2, in order to catch SNPs that occur in duplicated regions. These SNPs show up as heterozygous calls in the output. The minimum alternate fraction for such calls was set at 0.4. The p-value cutoff was set at 0.001. SNPs from all genomes were called simultaneously, using the *--no-ewens-priors* and *--no-marginals* flags. The *--variant-input flag was used to provide Freebayes with the recoded SNP (UAG-to-UAA) positions as putative variants to call regardless of evidence*. Reads supporting SNPs were required to have a minimum mapping quality of 10 and a minimum base quality of 30. Mapping quality was not otherwise used to assess SNP likelihoods (*--use-mapping-quality* was disabled). We ran Freebayes as described above to generate a single VCF file containing all variants for all samples. This VCF file was then further analyzed and filtered before as described below, before generating the summarizing diagram Figure S5-3.

SNP Effect using snpEFF: SnpEff 2.0.5d (*34*) was used to annotate variants and to predict effects for called SNPs. First, the reference genome's annotated GenBank Record (GI:48994873) was used to create a genome database, and the VCF records were annotated for coding effects only.

Final SNP filtering: In addition to the Freebayes SNP identification criteria, we used additional metrics to filter out SNPs that could not be called with high confidence. This additional filtering helped to reduce the complexity of the relationship of variants across all sequenced genomes in order to plot Figure S5-3. Note that this filtering resulted in some low-evidence variants being temporarily ignored in the aggregate analysis. However, these were carefully triaged and identified in the process of generating the sequence annotation file for the final C321.ΔA strain.

i. All 'heterozygous' calls were filtered out, as these represent SNPs whose reads map to multiple locations in the genome.

ii. SNPs that were present in fewer than three samples and could not be called either present or absent in >20 strains due to poor coverage or read mapping quality were filtered out.

iii. SNPs were filtered out if they were covered by ≤ 20 reads with good mapping quality across all genomes.

iv. SNPs that could be called absent or present in fewer than three genomes were removed.

Structural variants using Pindel and Breakdancer: Pindel (*35*) and Breakdancer (*36*) were both used to find potential structural variants in the genomes. First, Picard <http://picard.sourceforge.net/> was used to gather insert size metrics per genome. This information, along with the aligned BAM data, was run through Pindel. The Pindel output was converted to VCF using the *pindel2vcf* tool. We required at least 20 reads to support a breakpoint or junction. The *breakdancer_max* program in Breakdancer was also used to find structural variants. For Breakdancer, at least 8 read pairs were required to support a called structural event.

We manually corroborated structural variant calls from Pindel and Breakdancer through visual examination of read alignments. Since we observed a high-rate of false-positive and false-negative calls with these toolswe did not include them in our final strain analysis in the main text. Still, the Pindel and Breakdancer data were useful in troubleshooting cassette insertions and intentional gene knockouts and replacements.

Future work to combine evidence from these and additional tools might lead to a more robust, comprehensive, and high throughput method to validate structural variants using only short-read sequencing data.

Breakdancer predicted 49 unique events, and 187 total events across 69 strains. Because Breakdancer cannot call across multiple strains simultaneously and only gives approximate event locations based on read-pair distances, events that occurred in multiple samples were identified by using similar event start and end locations. Breakdancer predicted a total of 21 unique deletions, 5 unique inversions, and 23 unique translocations.

Pindel used split read data to predict both uncharacterized breakpoints and whole structural events. 258 unique uncharacterized breakpoints were found; 230 of these occur in only a single sample. Pindel also predicted 79 unique structural events. 9 were large deletions, 59 were insertions of unknown size, and 11 were inversions.

Coverage analysis: Coverage for each genome was analyzed using the bedtools (*37*) programs *makewindows* and *multicov.* The genome was split into 50 bp windows and BAM coverage was assessed for each window. A custom python script was used to take this information and find contiguous windows of low and high coverage, indicative of gene amplifications and deletions. These results are included as supplemental Table S5-31.

Confirming cassette insertion sites: Known insertion sites of CAGE antibiotic resistance markers were confirmed by selecting the reads that were soft clipped and/or not aligned to the MG1655, and aligning them to the known cassette sequences using Bowtie. Cassette insertion locations were inferred using the alignment locations of paired reads in which one read mapped to a cassette and the other mapped to a location on the genome.

Visually confirming SNPs and structural variants: The *tview* tool in the *Samtools* package (*38*) was used to visually inspect individual UAG SNPs and to assess the veracity of low-confidence SNP and structural variant calls.

Generating genome figures: Figure S5-3 was created using custom software written in R and Processing.

Fitness analysis: To assess fitness, strains were grown in flat-bottom 96-well plates (150 µL LB^L, 34 °C, 300 rpm). Kinetic growth (OD_{600}) was monitored on a Biotek H4 plate reader at 5 minute intervals. Doubling times were calculated by $t_{double} = c*\ln(2)/m$, where c = 5 minutes per time point and m is the maximum slope of $\ln(OD_{600})$. Since some strains achieved lower maximum cell densities, slope was calculated based on the linear regression of $\ln(OD_{600})$ through 5 contiguous time points (20 minutes) rather than between two pre-determined OD_{600} values. To monitor fitness changes in the CAGE lineage, growth curves were measured in triplicate, and their average was reported in Figure 5-2 and Table S5-1. To determine the effect of RF1 removal and NSAA incorporation on the panel of recoded strains (Table 5-1), growth curves were measured in triplicate (Figure 5-3A, Figure S5-8). Statistics were based on a Kruskal-Wallis one-way ANOVA followed by Dunn's multiple comparison test, where $*p < 0.05$, $**p < 0.01$, and $***p < 0.001$.

To assess re-growth phenotypes from long-term NSAA expression, overnight cultures were first grown in LB^L supplemented with chloramphenicol to maintain the pEVOL plasmids. These cultures were passaged into LB^L containing chloramphenicol, arabinose (to induce pEVOL), and either pAcF or pAzF depending on whether pEVOL-pAcF or pEVOL-pCNF was used. Growth with shaking at 34°C was monitored using a Biotek H1 or a Biotek Eon plate reader with OD_{600} readings every 10 minutes (pAcF) or 5 minutes (pAzF). After 16 hours of growth, the expression cultures were passaged into identical expression conditions and the growth curves were monitored with the same protocols.

NSAA incorporation assays:

Plasmids and strains for NSAA incorporation: p-acetyl-L-phenylalanine (pAcF) incorporation was achieved using pEVOL-pAcF (*9*) which contains two copies of pAcF-RS and one copy of $tRNA_{CUA}^{opt}$. The pEVOL-pAcF plasmid was maintained using chloramphenicol resistance. One copy of pAcF-RS and $tRNA_{CUA}^{opt}$ were constitutively expressed, and the second copy of pAcF-RS was under araBAD-inducible control (0.2% L-arabinose).

O-phospho-L-serine (Sep) incorporation was achieved by expression of $tRNA^{Sep}$ from pSepT and both EFSep (EF-Tu variant capable of incorporating Sep) and SepRS from pKD-SepRS-EFSep (*21*). To prevent enzymatic dephosphorylation of Sep *in vivo*, the gene encoding phosphoserine phosphatase (*serB*), which catalyzes the last step in serine biosynthesis, was inactivated. Specifically, Glu93 (GAA) was mutated to a premature UAA stop codon *via* MAGE. The pKD-SepRS-EFSep plasmid was maintained using kanamycin resistance and both SepRS and EFSep were induced using IPTG. The pSepT plasmid was maintained using tetracycline resistance, and $tRNA^{Sep}$ was constitutively expressed.

Effect of RF1 deletion, aaRS expression, and NSAA incorporation on fitness: Stationary phase pre-cultures were obtained by overnight growth with shaking at 34 °C in 150 µl LB^L supplemented with chloramphenicol for plasmid maintenance. Stationary phase cultures were diluted 100-fold into 150 µl LB^L containing chloramphenicol and 0.2% L-arabinose and/or 1 mM pAcF where indicated. Growth was monitored on a Biotek Synergy H1 plate reader. OD_{600} was recorded at 10-minute intervals for 16 hours at 34 °C with continuous shaking. All data were measured in triplicate. Doubling time was determined for each replicate as described above, and replicates were averaged for Figure 5-3A.

GFP variant synthesis: GFP variants (Table S5-33) were synthesized as gBlocks by IDT and modified with an N-terminal 6His tag *via* PCR. His-tagged GFP variants were isothermally assembled (*39*) into the pZE21 plasmid backbone (*40*) to yield the array of GFP reporter plasmids used in this study. Reporter plasmids were maintained using kanamycin resistance and induced using 30 ng/mL anhydrotetracycline (aTc).

UAG suppression and GFP Fluorescence: Stationary phase pre-cultures were obtained by overnight growth with shaking at 34 °C in 150 µl LB^L supplemented with appropriate antibiotics for plasmid maintenance. Stationary phase cultures were diluted 100-fold into 150 µl fresh LB^L containing the same antibiotics as the overnight pre-culture. These cultures were grown to mid-log phase and diluted 100-fold into 150 µl fresh LB^L containing the same antibiotics plus 30 ng/ml aTc, 0.2% L-arabinose, and/or 1 mM pAcF (where indicated). Protein expression proceeded for 16 hours at 34 °C with continuous shaking. Following 16 hours of expression, cultures were transferred to V-bottomed plates, pelleted, and washed once in 150 µL of PBS (pH 7.4). Washed pellets were resuspended in 150 µL of PBS (pH 7.4) and transferred to a black-walled, clear-bottom plate to measure GFP fluorescence for each strain. Both OD_{600} and

GFP fluorescence (Ex: 485 nm, Em: 528 nm) were measured on a Biotek Synergy H1 plate reader. Fluorescence and OD_{600} measurements were corrected by subtracting background fluorescence and OD_{600} (determined using PBS blanks). Relative fluorescence (in rfu) was calculated by the ratio fluorescence/OD_{600}. Reported values represent an average of four replicates. After measurements were complete, the cells were pelleted, the supernatant was aspirated, and the pellets were frozen at -80 °C for subsequent protein purification and Western blot analysis.

Protein extraction and Western blots: Cell pellets were obtained as described above. Cells were lysed using a lysis cocktail containing 150 mM NaCl, 50 mM Tris-HCl, 0.5x BugBuster reagent, 5% glycerol, 50 mM Na_3VO_4, 50 mM NaF, protease inhibitors (Roche), and 1 mM DTT. The resulting lysates were spun at 4 °C for 15 minutes at 3200 x g only in cases where soluble and insoluble fractions were separately analyzed. Protein lysate concentrations were determined using the BioRad-DC colormetric protein assay. Lysates were normalized by optical density at 600 nm, resolved by SDS-PAGE, and electro-blotted onto PVDF membranes (Millipore, # ISEQ00010). Western blot analysis was performed with mouse monoclonal antibody directed against GFP (Invitrogen, # 332600), and membranes were imaged with an HRP secondary antibody (Jackson Immunoresearch, JAC-715035150) *via* chemiluminescence on a ChemiDoc system (BioRad).

Mass spectrometry:

Materials: Urea, Tris-HCl, $CaCl_2$, iodoacetamide (IAA), Pyrrolidine, DL-lactic acid, HPLC grade water and acetonitrile (ACN) were from Sigma-Aldrich (St. Louis, MO). Chloroform and dithiothretitol (DTT) were from American Bioanalytical (Natick, MA). Methanol, trifluoroacetic acid (TFA), ammonium hydroxide and formic acid (FA) were obtained

from Burdick and Jackson (Morristown, NH). Sequencing grade modified trypsin was from Promega (Madison,WI). Anionic acid cleavable surfactant II (ALS) was from Protea (Morgantown, WV). UltraMicroSpin^{TM} columns, both the C_{18} and the DEAE PolyWAX variety were from The Nest Group, Inc. (Southborough, MA). Titaniumdioxide (TiO_2) with a particle size of 5 µm was obtained from GL Sciences Inc. (Torrance, CA).

Cell culture and lysis: Strains were routinely grown in LB^L media with the following concentration of antibiotics when appropriate: tetracycline (12 µg/mL), kanamycin (50 µg/mL), chloramphenicol (12 µg/mL), and zeocin (25 µg/mL). Bacterial cell cultures were grown at 30°C while shaking at 230 rpm until late log phase, quenched on ice and pelleted at 10,000 x g (10 min). The media was discarded and the cell pellets were frozen at -80°C to assist with subsequent protein extraction. Frozen cell pellets were thawed on ice and lysed in lysis buffer consisting of BugBuster reagent, 50 mM Tris-HCl (pH 7.4, 23°C), 500 mM NaCl, 0.5 mM EGTA, 0.5 mM EDTA, 14.3 mM 2-mercaptoethanol, 10 % glycerol, 50 mM NaF, and 1 mM Na_3O_4V, Phosphatase inhibitor cocktail 3 and complete protease inhibitor cocktail (Sigma Aldrich) were added as recommended by the corresponding manufacturer. Cell suspensions were incubated on ice for 30 min and the supernatant was removed after ultracentrifugation. The remaining pellet was re-extracted and resulting fractions were combined.

Protein lysates: Protein was precipitated with the methanol/chloroform method as previously described (*41*). One third of the resulting protein pellet was dissolved in 1.5 ml freshly prepared 8 M Urea/0.4 M Tris-HCl buffer (pH= 8.0, 23 °C). 5 mg protein was reduced and alkylated with IAA and digested overnight at 37°C using sequencing grade trypsin. The protein digest was desalted using C_{18} Sep-Pak (Waters) and the purified peptides were lyophilized and stored at -80°C.

Digestion of intact E. coli for shotgun proteomics: Cells were grown overnight to stationary phase, quenched on ice, and 2 ml culture was used for protein extraction and mass spectrometry. Cells were pelleted for 2 min at 2000 x g and the resulting pellet was washed twice with 1 ml ice cold Tris-HCl buffer pH=7.4, 23°C. The cells were then re-suspended in 100 µl Tris-HCl buffer pH=7.4, 23°C, split into 4 equal aliquots of 25 ul and the cell pellet was frozen at -80 °C. Frozen pellets were lysed with 40 µl lysis buffer consisting of 10 mM Tris-HCl buffer pH = 8.6 (23°C) supplemented with 10 mM DTT, 1 mM EDTA and 0.5 % ALS. Cells were lysed by vortex for 30 s and disulfide bonds were reduced by incubating the reaction for 35 min. at 55 °C in a heating block. The reaction was briefly quenched on ice and 16 µl of a 60 mM IAA solution was added. Alkylation of cysteines proceeded for 30 min in the dark. Excess IAA was quenched with 14 µl of a 25 mM DTT solution and the sample was then diluted with 330 µl of 183 mM Tris-HCl buffer pH=8.0 (23 °C) supplemented with 2 mM $CaCl_2$. Proteins were digested overnight using 12 µg sequencing grade trypsin for each protein aliquot, and the reaction was then quenched with 64 µl of a 20 % TFA solution, resulting in a sample pH<3. Remaining ALS reagent was cleaved for 15 min at room temperature. An aliquot of the sample consisting of ~30 µg protein (as determined by UV_{280} on a nanodrop) was desalted by reverse phase clean-up using C_{18} UltraMicroSpin columns. The desalted peptides were dried at room temperature in a rotary vacuum centrifuge and reconstituted in 30 µl 70 % formic acid 0.1 % TFA (3:8 v/v) for peptide quantitation by UV_{280}. The sample was diluted to a final concentration of 0.6 µg/µl and 4 µl (2.4 µg) were injected for LC-MS/MS analysis of the unfractionated digest using a 200 min method.

Phosphopeptide enrichment: Offline phosphopeptide enrichment was carried out with Titanium dioxide (TiO_2) using a bulk enrichment strategy adapted from Kettenbach *(42)*. Briefly,

between 0.4 and 1 mg of desalted peptide digest was transferred into a 1.5 ml PCR tube and dissolved at a concentration of 1mg/ml in “binding solution” consisting of 2 M lactic acid in 50 % ACN. Activated TiO_2 was prepared as a concentrated slurry in binding solution and added to the peptide solution to obtain a TiO_2 to peptide ratio of 4:1 by mass. The mixture was incubated for 2 h at room temperature on an Orbit M60 laboratory shaker operated at 140 rpm. The suspension was centrifuged for 20 s at 600 x g and the supernatant was removed. The TiO_2 beads were washed twice with 50 µl of the binding solution and then 3 times with 100 µl 50 % ACN, 0.1 % TFA. Stepwise elution of phosphopeptides from the beads was carried out using 20 µl of 0.2 M sodium phosphate buffer pH=7.8 followed by 20 µl 5 % ammonium hydroxide and 20 µl 5 % pyrrolidine solution. The pH of the combined extracts was adjusted with 30 µl of ice cold 20 % TFA resulting in a sample pH <3.0. Peptides were desalted on C_{18} UltraMicroSpin columns as described above and the peptide concentration was estimated by UV_{280}.

Offline fractionation of tryptic digests: Offline electrostatic repulsion-hydrophilic interaction chromatography (ERLIC) *(43)* was performed on disposable DEAE PolyWAX UltraMicroSpin columns. Columns were activated as recommended by the manufacturer and then conditioned with 3 x 200 µl washes with 90 % ACN, 0.1 % acetic acid (buffer A). For this purpose, the columns were centrifuged for at 200 x g for 1 min at 4°C. The column was then loaded with 50 µg of a desalted peptide digest prepared in 25 µl buffer A, and the flow-through was collected. Stepwise elution of the peptides was carried out using brief centrifugation steps carried out for 30 s at 200 x g with 50 µl eluent unless noted otherwise. The elution steps consisted of the following volumetric mixtures of buffer A and buffer B (0.1 % formic acid in 30 % ACN): (1) 100:0 (2) 96:4 (3) 90:10 (4) 80:20 (5) 60:40 (6) 100 µl of 20:80 (7) 100 µl of 0:100. Additional elution steps consisted of: (8) 1 M triethylamine buffer adjusted with formic

acid to pH=2.0. (9) 0.2 % ammonia (10) 0.2 % ammonia and finally (11) 100 µl 70 % formic acid. The collected fractions were dried in a vacuum centrifuge and reconstituted in 15 µl solvent consisting of 3:8 by volume of 70 % formic acid and 0.1 % TFA. Fractions were analyzed by LC-MS/MS using a 400 min gradient.

Liquid chromatography and mass spectrometry: Capillary LC-MS was performed on an Orbitrap Velos mass spectrometer (Thermo Fisher Scientific) connected to a nanoAcquity UPLC (Waters, Milford, MA). Liquid chromatography was performed at 35 °C with a vented split setup consisting of a commercially available 180 µm x 20 mm C_{18} nanoAcquity UPLC trap column and a BEH130C18 Waters symmetry 75 µm ID x 250 mm capillary column packed with 5 and 1.7 µm particles respectively. Mobile phase A was 0.1 % formic acid (FA) and mobile phase B was 0.1 % FA in acetonitrile. The injection volume was 4-5 µl depending on the sample concentration. Up to 2.4 µg peptides were injected for each analysis. Peptides were trapped for 3 min in 1 % B with and a flow rate of 5 µl/min. Gradient elution was performed with 90, 200 and 400 min methods with a flow rate of 300 nl/min. Two blank injections were performed between samples to limit potential carryover between the runs. The gradient for the 90 min method was 1-12 % B over 2 min, 12-25 % B over 43 min, 25-50 % B over 20 min, followed by 6 min at 95 % B and column re-equilibration in 1 % B. The gradient for the 200 min was 1-10 % B over 2 min, 10-25 % B over 150 min, and 25-50 % B over 20 min, followed by 7 min at 95 % B and re-column equilibration at 1 % B. The gradient for the 400 min was 1 min in 1 % B, 1-7 % B over 2 min, 7-20 % B over 298 min, and 20-50 % B over 60 min, followed by a 1 min flow ramp to 95 % B. The column was flushed for 9 min using 95 % B and then re-equilibrated for 27 min at 1 % B prior to the next injection. Mass spectrometry was performed with a spray voltage of 1.8 kV

and a capillary temperature of 270 °C. A top 10 Higher Collisional Energy Dissociation (HCD) method with one precursor survey scan (300-1750 Da) and up to 10 tandem MS spectra performed with an isolation window of 2 Da and a normalized collision energy of 40 eV. The resolving power (at m/z = 400) of the Orbitrap was 30,000 for the precursor and 7500 for the fragment ion spectra, respectively. Continuous lock mass calibration was enabled using the polycyclodimethylsiloxane peak (m/z = 445.120025) as described *(44)*. Dynamic exclusion criteria were set to fragment precursor ions exceeding 3000 counts with a charge state >1 twice within a 30 s period before excluding them from subsequent analysis for a period of 60 s. The exclusion list size was 500 and early expiration was disabled.

Proteomics data processing: Raw files from the Orbitrap were processed with Mascot Distiller and searched in-house with MASCOT (v. 2.4.0) against the EcoCyc *(45)* protein database release 16.0 for *E. coli* K-12 substr. MG1655 with a custom database and search strategy designed to identify amber suppression (Aerni et al. manuscript in preparation). Forward and decoy database searches were performed with full trypsin specificity allowing up to 3 missed cleavages and using a mass tolerance of ±30 ppm for the precursor and ±0.1 Da for fragment ions, respectively. Cysteines were considered to be completely alkylated with IAA unless samples were processed by a gel-based workflow. In that case Propionamide (C) was considered as a variable modification. Additional variable modifications for all searches were oxidation (M) and deamidation (NQ) for samples processed with urea Carbamyl (K, R, N-term). In order to detect pAcF containing peptides, a variable custom modification for Y was introduced with the composition C_2H_2 and monoisotopic mass of 26.015650 Da. Typical FDR were <1 % for peptides above identity threshold and <2% considering all peptides above identity or homology threshold respectively. The MASCOT search results were deposited in the Yale Protein

Expression Database (YPED) *(46)*. The following filter rules were specified in YPED for reporting of protein identifications: (i) At least 2 bold peptides and peptide scores ≥20 or (ii) 1 bold red peptide with a peptide score ≥20 with at least one additional bold red peptide with a score between 15 and 20.

Bacteriophage assays: For all phage experiments, growth was carried out in LB^L at 30 °C. Liquid cultures were aerated with shaking at 300 rpm. Before each experiment, a fresh phage lysate was prepared. To do this, *Escherichia coli* MG1655 was grown to mid-log phase in 3 mL of LB^L, then ~2 uL of T7 bacteriophage (ATCC strain BAA-1025-B2) or T4 bacteriophage (ATCC strain 11303-B4) was added directly from a glycerol stock into the bacterial culture. Lysis proceeded until it was complete (lysate appears clear after ~4 hours). The entire lysate was centrifuged to remove cell debris (10,000 rcf, 10 minutes), and 3 mL of lysate was transferred to a glass vial supplemented with 150 mg NaCl for phage preservation. Lysates were prepared fresh, titered, and stored at 4 °C for the duration of each experiment. One lysate was used for all replicates of a given experiment.

Phage titering: Phage lysate was titered by serial dilution into LB^L (10-fold dilution series). Before plating on LB^L agar, 10 µL of the diluted phage lysate was mixed with 300 µL of mid-log *E. coli* MG1655 culture and 3 mL of molten top agar. Plaques matured for ~4 hours at 30 °C. Titers (pfu/mL) were calculated based on the lysate dilutions that produced 20-200 pfu.

Plaque area: For plaque area assays, bacterial cultures were grown to mid-log phase in 3 mL LB^L. To accommodate different doubling times, faster-growing cultures were continually diluted until all strains reached OD_{600} ~0.5. Immediately prior to infection, OD_{600} was normalized to 0.50 for all cultures. Approximately 30 pfu of T7 bacteriophage were mixed with 300 µL of OD_{600} = 0.50 culture and 3 mL of molten top agar, and then immediately plated on

LB^L agar. Plaques were allowed to mature at 30 °C for 7 hours, then the plates were imaged on a Bio-Rad Gel Doc system, and plaque areas were measured using ImageJ (*47*). Statistics were based on a Kruskal-Wallis one-way ANOVA followed by Dunn's multiple comparison test, where *$p < 0.05$, **$p < 0.01$, and ***$p < 0.001$.

T7 Fitness: Fitness was assessed in triplicate at low MOI based on protocols by Heineman *et al.* (*22*). Briefly, bacterial glycerol stocks were inoculated directly into 3 mL LB^L and serially diluted in LB^L. Serial dilutions were grown overnight (30 °C, 300 rpm), so that one of the dilutions would be at mid-log growth phase in the morning. Prior to infection, a second dilution series was performed so that host strains would be at optimal growth phase over the course of the serial infection. Starting cultures were normalized to OD_{600} = 0.50 by adding LB^L immediately before infecting the cultures (MOI = 0.015) at t = 0. Infected culture was diluted 1/10 into 3 mL of uninfected mid-log phase culture at 30 minute intervals. Aliquots of the infection were taken at t = 4, 10, 60, and 120 minutes. At t = 4, the aliquot was treated with chloroform to quantitate non-adsorbed phage particles. For all other time points (t = 10, 60, and 120), aliquots were immediately mixed with 300 µL of mid-log *E. coli* MG1655 and 3 mL molten top agar and then spread on LB^L agar. Plaques were counted after maturing for ~4 hours at 30 °C, and then pfu/mL was calculated for each time point, correcting for dilutions. Adsorption efficiency was consistently >95% as determined by $(N_{t=4} - N_{t=10}) / N_{t=10}$, and fitness was determined by $[\log_2(N_{t=120}/N_{t=60})]/(\Delta t/(60 \text{ min/hr}))$, where N is the number of phages at time t minutes and $\Delta t = 60$ min.

Kinetic lysis time: Mean lysis time was determined with 12 replicates based on protocols from Heineman *et al.* (*22*), except that OD_{600} was monitored instead of OD_{540}. Mid-log phase

cells (as in the fitness assay) were infected at MOI = 5, then 150 µL aliquots of infected culture were distributed into a 96-well flat bottomed plate and sealed with Breathe-Easy™ sealing membrane. Lysis was monitored at 30 °C with shaking at 300 rpm on a Biotek H4 plate reader with OD_{600} measurements taken every 5 minutes. Each lysis curve was fit to a cumulative normal distribution using the normcdf function in MATLAB. Mean lysis time, mean lysis OD_{600}, and mean lysis slope were calculated using this cumulative normal distribution function.

Supplemental material

Supplemental material for CHAPTER 5 can be found in APPENDIX D or at <http://www.sciencemag.org/content/suppl/2013/10/16/342.6156.357.DC1/Lajoie.SM.pdf>. Additional supplemental tables can be found at <http://www.sciencemag.org/content/342/6156/357/suppl/DC1>.

References

1. K. Vetsigian, C. Woese, N. Goldenfeld, Collective evolution and the genetic code. *PNAS* **103**, 10696 (2006).

2. D. V. Goeddel *et al.*, Expression in Escherichia coli of Chemically Synthesized Genes for Human Insulin. *PNAS* **76**, 106 (1979).

3. D. C. Krakauer, V. A. A. Jansen, Red queen dynamics of protein translation. *J. Theor. Biol.* **218**, 97 (2002).

4. M. G. Schafer *et al.*, The Establishment of Genetically Engineered Canola Populations in the U.S. *PLoS One* **6**, e25736 (2011).

5. J. M. Sturino, T. R. Klaenhammer, Engineered bacteriophage-defence systems in bioprocessing. *Nat. Rev. Microbiol.* **4**, 395 (2006).

6. M. Schmidt, V. de Lorenzo, Synthetic constructs in/for the environment: Managing the interplay between natural and engineered Biology. *FEBS Lett.* **586**, 2199 (2012).

7. C. C. Liu, P. G. Schultz, Adding New Chemistries to the Genetic Code. *An. Rev. Biochem.* **79**, 413 (2010).

8. H. Neumann, K. Wang, L. Davis, M. Garcia-Alai, J. W. Chin, Encoding multiple unnatural amino acids via evolution of a quadruplet-decoding ribosome. *Nature* **464**, 441 (2010).

9. T. S. Young, I. Ahmad, J. A. Yin, P. G. Schultz, An Enhanced System for Unnatural Amino Acid Mutagenesis in E. coli. *Journal of Molecular Biology* **395**, 361 (2009).

10. G. Eggertsson, D. Söll, Transfer ribonucleic acid-mediated suppression of termination codons in Escherichia coli. *Microbiological Reviews* **52**, 354 (September 1, 1988, 1988).

11. F. J. Isaacs *et al.*, Precise Manipulation of Chromosomes in Vivo Enables Genome-Wide Codon Replacement. *Science* **333**, 348 (Jul, 2011).

12. D. G. Gibson *et al.*, Creation of a Bacterial Cell Controlled by a Chemically Synthesized Genome. *Science* **329**, 52 (Jul, 2010).

13. H. H. Wang *et al.*, Programming cells by multiplex genome engineering and accelerated evolution. *Nature* **460**, 894 (Aug, 2009).

14. P. A. Carr *et al.*, Enhanced multiplex genome engineering through co-operative oligonucleotide co-selection. *Nucleic Acids Res.*, (May 25, 2012, 2012).

15. T. Mukai *et al.*, Codon reassignment in the Escherichia coli genetic code. *Nucleic Acids Res.* **38**, 8188 (2010).

16. D. B. F. Johnson *et al.*, RF1 knockout allows ribosomal incorporation of unnatural amino acids at multiple sites. *Nat Chem Biol* **7**, 779 (2011).

17. K. Ohtake *et al.*, Efficient Decoding of the UAG Triplet as a Full-Fledged Sense Codon Enhances the Growth of a prfA-Deficient Strain of Escherichia coli. *J. Bacteriol.* **194**, 2606 (May 15, 2012, 2012).

18. P. O'Donoghue *et al.*, Near-cognate suppression of amber, opal and quadruplet codons competes with aminoacyl-tRNAPyl for genetic code expansion. *FEBS Lett.*, (2012).

19. I. U. Heinemann *et al.*, Enhanced phosphoserine insertion during Escherichia coli protein synthesis via partial UAG codon reassignment and release factor 1 deletion. *FEBS Lett.* **586**, 3716 (2012-Oct-19, 2012).

20. J. T. Ngo, D. A. Tirrell, Noncanonical amino acids in the interrogation of cellular protein synthesis. *Accounts of chemical research* **44**, 677 (2011).

21. H.-S. Park *et al.*, Expanding the Genetic Code of Escherichia coli with Phosphoserine. *Science* **333**, 1151 (August 26, 2011, 2011).

22. R. H. Heineman, I. J. Molineux, J. J. Bull, Evolutionary robustness of an optimal phenotype: Re-evolution of lysis in a bacteriophage deleted for its lysin gene. *J. Mol. Evol.* **61**, 181 (Aug, 2005).

23. J. D. Bain, C. Switzer, R. Chamberlin, S. A. Benner, Ribosome-mediated incorporation of a nonstandard amino acid into a peptide through expansion of the genetic code. *Nature* **356**, 537 (APR 9 1992, 1992).

24. J. C. Anderson *et al.*, An expanded genetic code with a functional quadruplet codon. *Proc. Natl. Acad. Sci. U. S. A.* **101**, 7566 (May 18, 2004, 2004).

25. M. J. Lajoie *et al.*, Probing the Limits of Genetic Recoding in Essential Genes. *Science* **342**, 361 (Oct 18, 2013).

26. S. A. Schwartz, D. R. Helinski, Purification and Characterization of Colicin E1. *J. Biol. Chem.* **246**, 6318 (October 25, 1971, 1971).

27. J. A. Mosberg, M. J. Lajoie, G. M. Church, Lambda Red Recombineering in Escherichia coli Occurs Through a Fully Single-Stranded Intermediate. *Genetics* **186**, 791 (Nov, 2010).

28. D. Figurski, R. Meyer, D. S. Miller, D. R. Helinski, Generation in vitro of deletions in the broad host range plasmid RK2 using phage Mu insertions and a restriction endonuclease. *Gene* **1**, 107 (1976).

29. S. Warming, N. Costantino, D. L. Court, N. A. Jenkins, N. G. Copeland, Simple and highly efficient BAC recombineering using galK selection. *Nucleic Acids Res.* **33**, e36 (2005).

30. N. Rohland, D. Reich, Cost-effective, high-throughput DNA sequencing libraries for multiplexed target capture. *Genome Research* **22**, 939 (May, 2012).

31. W. R. Pearson, T. Wood, Z. Zhang, W. Miller, Comparison of DNA sequences with protein sequences. *Genomics* **46**, 24 (Nov, 1997).

32. B. Langmead, S. L. Salzberg, Fast gapped-read alignment with Bowtie 2. *Nat. Methods* **9**, 357 (Apr, 2012).

33. M. A. DePristo, E. Banks, R. Poplin, K. V. Garimella, J. R. Maguire, C. Hartl, A. A. Philippakis, G. del Angel, M. A. Rivas, M. Hanna, A. McKenna, T. J. Fennell, A. M. Kernytsky, A. Y. Sivachenko, K. Cibulskis, S. B. Gabriel, D. Altshuler, M. J. Daly, A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics* **43**, 491 (May, 2011).

34. P. Cingolani, A. Platts, L. L. Wang, M. Coon, N. Tung, L. Wang, S. J. Land, X. Lu, D. M. Ruden, A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w(1118); iso-2; iso-3. *Fly* **6**, 80 (Apr-Jun, 2012).

35. K. Ye, M. H. Schulz, Q. Long, R. Apweiler, Z. Ning, Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics* **25**, 2865 (Nov 1, 2009).

36. K. Chen, J. W. Wallis, M. D. McLellan, D. E. Larson, J. M. Kalicki, C. S. Pohl, S. D. McGrath, M. C. Wendl, Q. Zhang, D. P. Locke, X. Shi, R. S. Fulton, T. J. Ley, R. K. Wilson, L. Ding, E. R. Mardis, BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. *Nat. Methods* **6**, 677 (Sep, 2009).

37. A. R. Quinlan, I. M. Hall, BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26**, 841 (Mar 15, 2010).

38. H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin, P. Genome Project Data, The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**, 2078 (Aug, 2009).

39. D. G. Gibson, H. O. Smith, C. A. Hutchison, J. C. Venter, C. Merryman, Chemical synthesis of the mouse mitochondrial genome. *Nat. Methods* **7**, 901 (Nov, 2010).

40. R. Lutz, H. Bujard, Independent and Tight Regulation of Transcriptional Units in Escherichia Coli Via the LacR/O, the TetR/O and AraC/I1-I2 Regulatory Elements. *Nucleic Acids Res.* **25**, 1203 (March 1, 1997, 1997).

41. D. Wessel, U. I. Flugge, A Method for the Quantitative Recovery of Protein in Dilute-Solution in the Presence of Detergents and Lipids. *Anal. Biochem.* **138**, 141 (1984).

42. A. N. Kettenbach, S. A. Gerber, Rapid and reproducible single-stage phosphopeptide enrichment of complex peptide mixtures: application to general and phosphotyrosine-specific phosphoproteomics experiments. *Anal. Chem.* **83**, 7635 (Oct 15, 2011).

43. A. J. Alpert, Electrostatic repulsion hydrophilic interaction chromatography for isocratic separation of charged solutes and selective isolation of phosphopeptides. *Anal. Chem.* **80**, 62 (Jan 1, 2008).

44. J. V. Olsen, L. M. de Godoy, G. Li, B. Macek, P. Mortensen, R. Pesch, A. Makarov, O. Lange, S. Horning, M. Mann, Parts per Million Mass Accuracy on an Orbitrap Mass Spectrometer via Lock Mass Injection into a C-trap. *Mol Cell Proteomics* **4**, 2010 (Dec, 2005).

45. I. M. Keseler, J. Collado-Vides, A. Santos-Zavaleta, M. Peralta-Gil, S. Gama-Castro, L. Muñiz-Rascado, C. Bonavides-Martinez, S. Paley, M. Krummenacker, T. Altman, P. Kaipa, A. Spaulding, J. Pacheco, M. Latendresse, C. Fulcher, M. Sarker, A. G. Shearer, A. Mackie, I. Paulsen, R. P. Gunsalus, P. D. Karp, EcoCyc: a comprehensive database of Escherichia coli biology. *Nucleic Acids Res.* **39**, D583 (January 1, 2011, 2011).

46. M. A. Shifman, Y. Li, C. M. Colangelo, K. L. Stone, T. L. Wu, K.-H. Cheung, P. L. Miller, K. R. Williams, YPED: A Web-Accessible Database System for Protein Expression Analysis. *Journal of Proteome Research* **6**, 4019 (2007/10/01, 2007).

47. C. A. Schneider, W. S. Rasband, K. W. Eliceiri, NIH Image to ImageJ: 25 years of image analysis. *Nat Meth* **9**, 671 (2012).

CHAPTER 6

Probing the Limits of Genetic Recoding in Essential Genes

This chapter is reproduced with minor edits with permission from its initial publication:

Lajoie MJ[*], Kosuri S[*], Mosberg JA, Gregg CJ, Zhang D, Church GM (2013) *Probing the limits of genetic recoding in essential genes*. **Science** 342: 361-3.

Research contributions: S. Kosuri and G. Church conceived of the project. S. Kosuri and D. Zhang performed gene assembly from OLS pools. M. Lajoie, J. Mosberg, and C. Gregg performed recombinations, genotype validation, and fitness assessment. M. Lajoie and S. Kosuri planned the experiments and analyzed the results. M. Lajoie, S. Kosuri, and G. Church oversaw all aspects of the project.

Acknowledgements: We thank Sara Vassallo and Joanne Ho for technical assistance; Uri Laserson, Dan Goodman, Nikolai Eroshenko, Dan Mandell, Dieter Söll, Lanny Ling, and Farren Isaacs for helpful comments. Funding was from Department of Energy [DE-FG02-02ER63445], NSF [SA5283-11210], DARPA [N66001-12-C-4040], U.S. Office of Naval Research [N000141010144], Agilent Technologies, Wyss Institute, and Department of Defense NDSEG Fellowship (M.J.L.).

Abstract

Engineering radically altered genetic codes will allow for genomically recoded organisms that have expanded chemical capabilities and are isolated from nature. We have previously reassigned the translation function of the UAG stop codon; however, reassigning sense codons poses a greater challenge because such codons are more prevalent, and their usage regulates gene expression in ways that are difficult to predict. To assess the feasibility of radically altering the genetic code, we selected a panel of 42 highly-expressed essential genes for modification. Across 80 *Escherichia coli* strains, we removed all instances of 13 rare codons from these genes and attempted to shuffle all remaining codons. Our results suggest that the genome-wide removal of 13 codons is feasible; however, several genome design constraints were apparent, underscoring the importance of a strategy that rapidly prototypes and tests many designs in small pieces.

Introduction

The canonical genetic code is nearly universal (*1*), allowing natural organisms to share beneficial traits via horizontal gene transfer. Genetically modified organisms also share this code, rendering them susceptible to viruses and capable of releasing recombinant genetic material (*e.g.* resistance genes (*2*)) into the environment. By redefining the genetic code, we hope to produce genomically recoded organisms (GROs) that are safe and useful.

In separate work, we have completely reassigned the UAG codon in *Escherichia coli* MG1655 (*3*) UAG was chosen for its rarity and simplicity of function, but our results (*3*) reinforce that sense codons must also be reassigned to achieve robust genetic isolation, broad virus resistance, and expanded chemical versatility (*4*). However, sense codon reassignment poses a considerable challenge given that codon usage can strongly affect gene regulation (*5*),

ribosome spacing (*6, 7*), translation efficiency (*7, 8*), translation levels (*9*), translation accuracy (*10*), and protein folding (*11, 12*). Furthermore, DNA/RNA motifs can provide additional noncoding functions such as regulating translation initiation via 5′ mRNA secondary structure (*13*), sharing sequence with overlapping small RNAs (*14*), pausing the ribosome at internal Shine-Dalgarno sequences (*15*), and regulating mRNA localization (*16*). Therefore, it is difficult to predict the effects of a given codon change, and these factors may substantially constrain the malleability of the genome. However, despite the myriad mechanisms by which swapping synonymous codons could be deleterious, efforts to express a codon-randomized *Klebsiella* nitrogenase gene cluster in *E. coli* have been successful, albeit with reduced activity compared with wild-type (*16*).

Although such information is critical for reassigning the genetic code, genome-wide codon essentiality has largely been unexplored, perhaps due to the substantial degree of genetic modification necessary for addressing such questions. For example, the complete removal of 13 codons corresponding to the least frequently used anticodons (Figure 6-1A and supplemental text) will require 155,224 changes in *E. coli* MG1655, several of which may not be tolerated. Although it has never been attempted, *de novo* genome design, synthesis, and transplantation (*17*) seems unlikely to produce a viable genome bearing this unprecedented number of potentially deleterious changes. Indeed, lethal genetic elements have been difficult to identify and eliminate using *de novo* genome transplantation (*17*). Therefore, we have developed *in vivo* multiplex genome editing technologies (*18, 19*) to rapidly prototype and manufacture genomes. Our approach exploits diversity and natural selection, and is highly amenable to our goal of testing the flexibility of synonymous codon choices as they pertain to reassigning the genetic code.

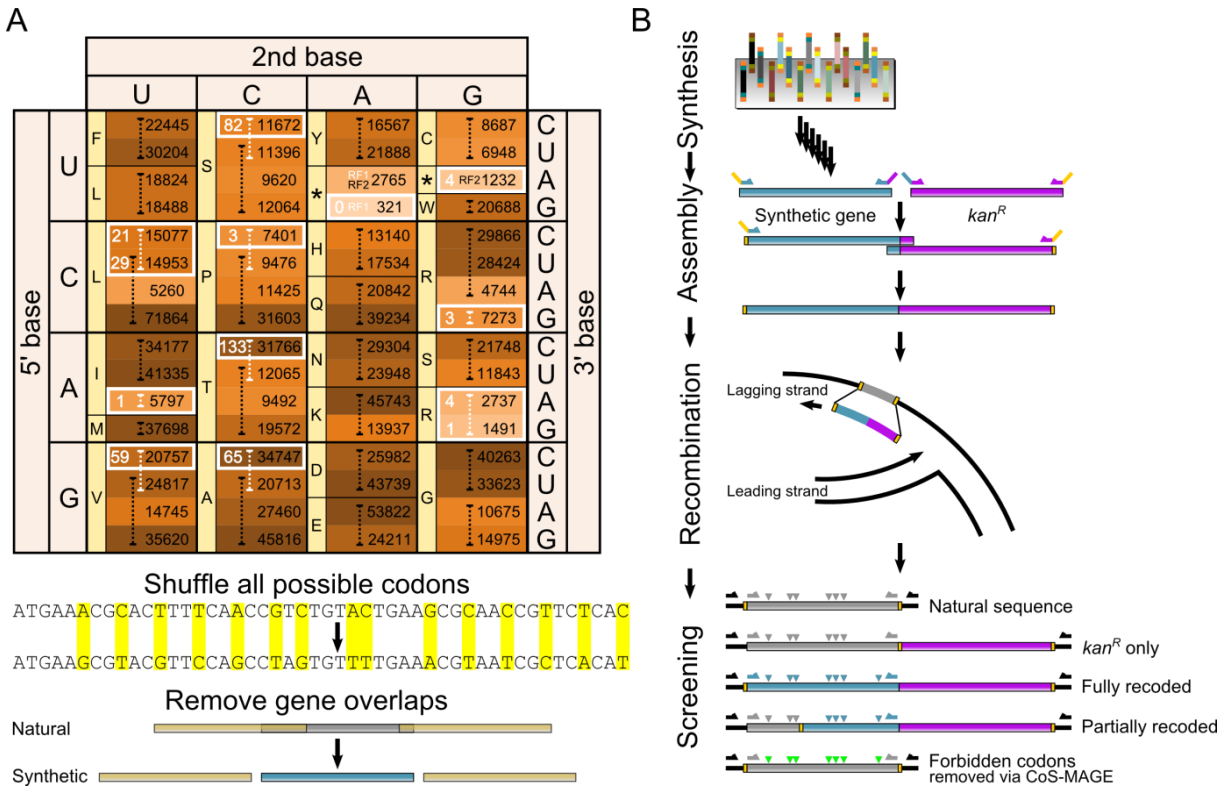


Figure 6-1. Codon reassignment across 42 essential genes. **(A)** *E. coli* MG1655 codon usage heat map; brightness increases as codon usage decreases. Black numbers are total codon usage based on NC_000913.2 (National Center for Biotechnology Information, 1 September 2011). The anticodon specificities (*29, 30*) are illustrated as dashed brackets; white indicates anticodons that were targeted for eventual removal. Amino acids are indicated in the yellow side bars. White boxes denote the 13 forbidden codons, and white numbers report how many instances of each codon were in the panel of 42 targeted essential genes. All 405 instances of these forbidden codons were successfully recoded across 80 *E. coli* strains. Additionally, all possible codons were swapped to synonymous codons, and gene overlaps were removed by duplication (bottom). **(B)** Strategy for recoding essential genes. Recoded genes (blue rectangles) were synthesized from Agilent oligonucleotide library synthesis arrays (*24*), then transcriptionally fused to *kan*^{R} (purple rectangles) by isothermal assembly (*25*). These cassettes were recombined into EcNR2 {*E. coli* MG1655 Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*] Δ*mutS*::*cat*} using λ Red (*26*), and recombinants were selected on kanamycin. Putative recombinants were screened with three sets of primers: wild-type primers (gray) hybridize specifically to the natural gene sequence, mutant primers (blue) hybridize specifically to the recoded gene sequence, and boundary primers (black) hybridize to the surrounding genomic DNA. Desired recombinants were detected by polymerase chain reaction and then verified by Sanger sequencing. We found that *kan*^{R} ("*kan*^{R} only") could be inserted downstream of all genes except for *rplO* without causing major deleterious effects. We attempted to replace all 42 natural genes with radically recoded versions ("Fully recoded"; blue rectangles and triangles are recoded sequence). To coarsely map problematic design elements in the failed cassettes, we prepared cassettes that preserved natural sequence at the N-terminus ("Partially recoded"; gray rectangles and triangles are natural sequence). Finally, all remaining forbidden codons were recoded with CoS-MAGE (green triangles) and confirmed with Sanger sequencing.

Results

To test whether we could radically change codon usage in order to free up codons that could be reassigned to alternate translation functions, we attempted to individually recode 42

essential genes, including all 41 essential ribosomal protein-coding genes (*20*) and *prfB*, which relies on a programmed frameshift for proper translation (*21*). Because expression level correlates strongly with codon usage bias (*9*), the highly expressed and tightly regulated (*22, 23*) ribosomal genes should be among the most difficult to change. To study codon essentiality in each of these genes, we attempted to remove all instances of the aforementioned 13 codons (hereafter referred to as "forbidden" codons). In addition, we gauged tolerance for large-scale DNA sequence alterations by shuffling all possible codons to synonymous alternatives. Replacement codons were chosen randomly from a weighted distribution, based on their frequencies in all *E. coli* genes (AUG and UGG codons were unchanged because they uniquely encode Met and Trp, respectively). Finally, we changed 1 non-AUG start codon to AUG, separated six gene overlaps, removed one frameshift, and avoided the use of six restriction sites used in gene assembly (supplemental text). Thus, whereas the protein sequence was 100% identical in our designs, the nucleotide sequence was on average only 65.4% identical, and the codon identity was only 4.44% (corresponding to the unchanged AUG and UGG codons) (Table S6-1). Based on these radical design parameters, we did not expect all design elements to be tolerated. Therefore, individually recoding each gene was the most biologically relevant scale on which to assess the effects of recoding without sacrificing the ability to rapidly map design flaws.

We synthesized recoded genes from DNA microchips (*24*), transcriptionally fused each to a kanamycin resistance gene (kan^R) by isothermal assembly (*25*), and replaced the corresponding natural gene (one gene per strain) *in vivo* using λ Red recombination (*26*) (Figure 6-1B). We also introduced kan^R downstream of the natural genes and found that 41 of 42 (Table S6-2) allowed insertion with an average growth defect of 15% in LB-Lennox (12% in Teknova

Hi-Def Azure media). Insertion downstream of *rplO* was unsuccessful, indicating that disrupting operon structure—and, by extension, refactoring overlapping genes—is a potential failure mode for redesigning genomes. For the recoded genes, we found that 26 of 42 (Table S6-2) were successful with an average growth defect of 20% in LB-Lennox (14% in Azure) compared with *kan*^{R} insertion controls (Figure 6-2). In the recoded *prfB* strain, removing the frameshift and recoding an upstream AGG codon that may be involved in pausing translation and enhancing frameshifting (*15*) did not significantly affect fitness (t test, p = 0.86). Finally, to test the independence of the growth defects, we inserted a recoded *rplM* or *rpsI* gene transcriptionally fused to spectinomycin resistance into three recoded strains with varying fitness (rpmC_syn1, rplE_syn1, and rplP_syn1). All double-mutant strains exhibited better fitness than predicted assuming that the fitness defects were independent, although this does not rule out potential cumulative effects from combining multiple deleterious designs (Figure S6-1).

The 16 unsuccessfully recoded genes provided an opportunity to identify failure modes for recoding. We coarsely mapped deleterious alleles in the remaining genes by recoding only the C-terminal half of each gene (successful for 9 of 16 genes)

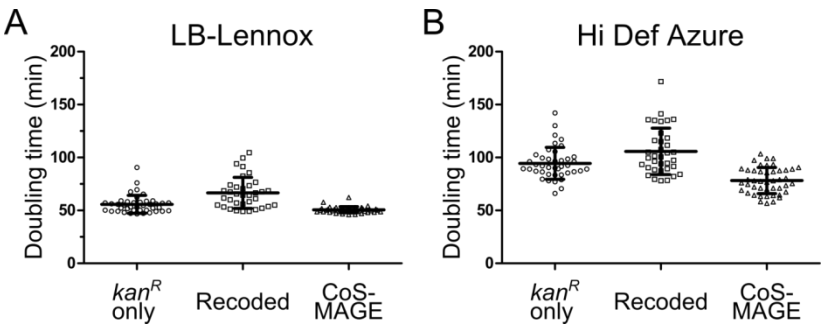


Figure 6-2. Recoded strain doubling times in **(A)** LB-Lennox media and **(B)** Teknova Hi-Def Azure media. Each data point represents the average doubling time of a given strain with a portion of a ribosomal gene recoded (n = 3). Error bars for each group represent mean +/- SD. Under assay conditions, the parental strain {*E. coli* MG1655 Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla* Δ*mutS*::*cat*], "EcNR2"} exhibited a 49 +/- 4 minute doubling time in LB-Lennox and a 84 +/- 5 minute doubling time in Teknova Hi-Def Azure. Strain genotypes and doubling times are summarized in Tables S6-4 to S6-5. *Kan*^{R} insertion into natural sequences (with no recoding) seldom impaired fitness. Still, we could not introduce *kan*^{R} downstream of *rplO* after three attempts. Fully or partially recoded gene recombinants exhibited the broadest range of fitness defects. For successful recombinants, position of the recoded gene in its operon did not appear to correlate strongly with fitness. The CoS-MAGE recombinants exhibited robust fitness, indicating that all tested forbidden codons are readily dispensable in small groups.

(Table S6-2). Of these 9 genes, 7 were also amenable to recoding all but the first 30 codons (Figure 6-1B). Although not conclusive based on the limited sample size, these remaining failed replacements may be caused by the disruption of endogenous control mechanisms upstream of the gene (*23*) or by codon bias affecting expression (*7, 12*). Using the above synthetic complementation approaches, we recoded a total of 294 of 405 forbidden codons in 35 of 42 targeted essential genes across 35 strains (one recoded gene per strain) (Tables S6-2 and S6-3). This generated 4375 out of 6496 total desired nucleotide changes and introduced 29 synthesis errors and/or spontaneous mutations (1 error per 436 base pairs) (Figure 6-3 and Table S6-4). Although synthesis errors sometimes introduced *de novo* forbidden codons, additional screening invariably found alternative clones lacking forbidden codons. We hypothesize that the remaining genes (7 of 16) failed due to perturbations in gene expression arising from separating overlapping genes, and/or non-viable changes introduced while shuffling codons that were not forbidden.

To determine whether any remaining instances of the forbidden codons were essential, we used co-selection multiplex automated genome engineering (CoS-MAGE) (*27*) to remove all remaining forbidden codons in small groups across a population of cells (111 desired mutations in 45 clones) (Figure 6-3 and Table S6-5). The CoS-MAGE recombinants exhibited robust fitness (Figure 6-2), indicating that none of the forbidden codons provide a systematic barrier to removal. Furthermore, this suggests that unsuccessful gene replacements using fully recoded cassettes were not due to the removal of forbidden codons. Our initial designs yielded all desired mutations except for one (*rplQ* U162G). Unexpectedly, when we attempted to replace this CUU (Leu) codon using a pool of oligos encoding all Leu, Ile, Val, and Ala codons (Table S6-6), only CUG (Leu), UUG (Leu), and GUG (Val) were not observed (Table S6-7). Therefore, CUU is not

essential, but 3 out of 12 tested replacement codons (all ending in UG) were either deleterious or recalcitrant to λ Red-mediated allele replacement in a way that was not anticipated *a priori*. We

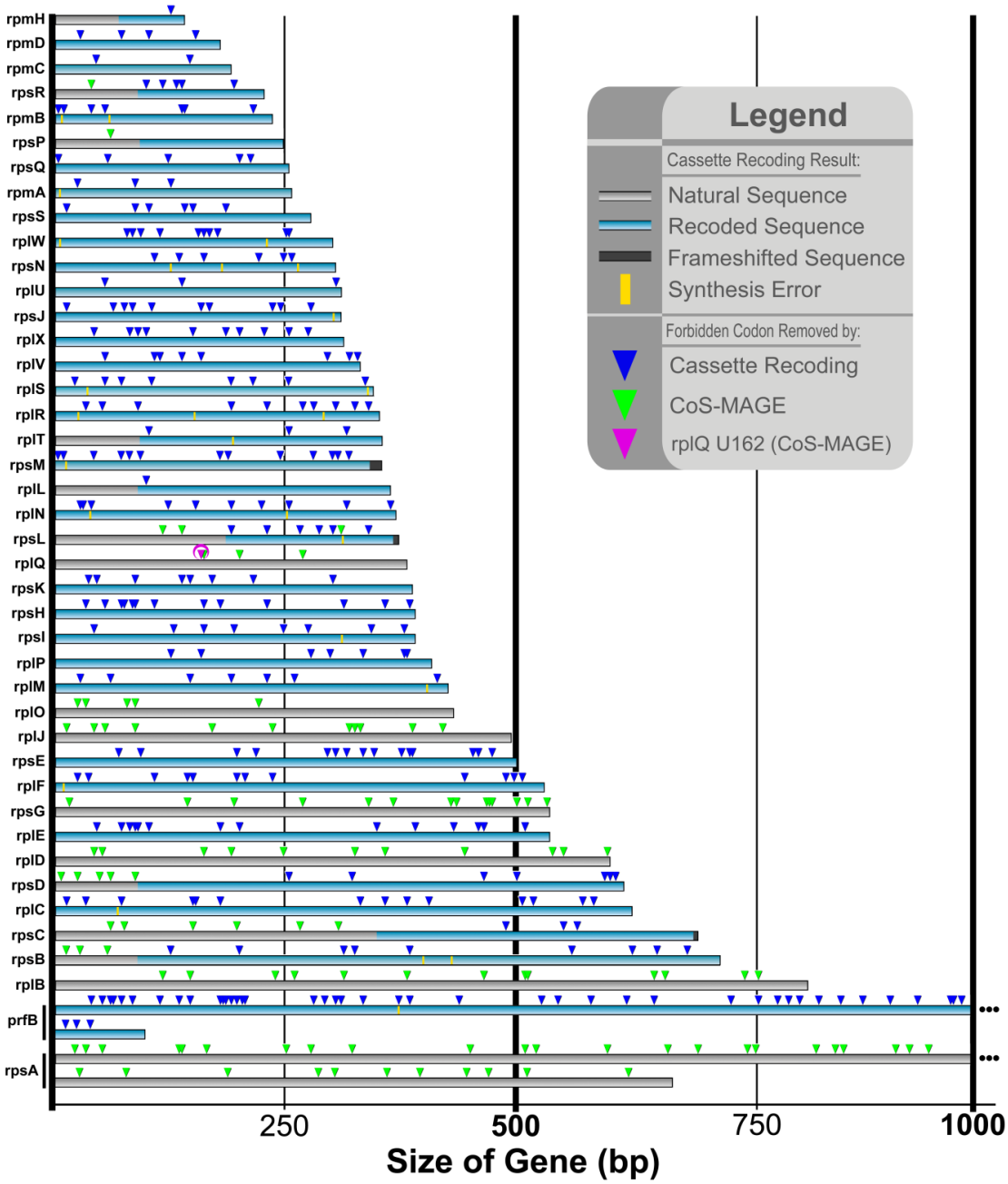


Figure 6-3. Schematics of all changes introduced in recoded essential genes. Light gray represents natural DNA sequence, light blue represents recoded sequence (average nucleotide identity = 65.4%), and dark gray represents frameshifted sequences caused by point deletions. Yellow lines indicate missense mutations introduced by gene synthesis errors, none of which introduced forbidden codons. Triangles indicate forbidden codons recoded by gene replacement (blue) or CoS-MAGE (green). The purple triangle in *rplQ* indicates the CUU codon that could not be converted to CUG as originally designed. We exhaustively tested all possible replacement Leu, Ile, Val, and Ala codons, and only CUG, UUG, and GUG were not observed. All 405 instances of the forbidden codons were successfully replaced across 80 strains.

note that the native gene sequence at this locus (ACT CTT GCC) contains most of the CTWGG Vsr recognition motif [Vsr is a mismatch repair endonuclease that is somewhat MutS-independent (*28*)], but that the position (nucleotide 3 instead of nucleotide 2) and identity (T:C instead of T:G) of the oligo-mediated mismatch are noncanonical. We mutated codons 23 and 24 of *vsr* to in-frame stop codons but were still unable to isolate *rplQ* recombinants with CUG, UUG, or GUG codons at position 162, thus suggesting that Vsr is not the cause of these failed replacements. It is likely that further recoding will uncover additional cryptic design flaws; nevertheless, our strategy is well suited to rapidly identify alternative solutions that are viable.

Discussion

Our results provide three important insights for designing recoded genomes. First, when tested individually or in groups, all 405 instances of the forbidden codons were non-essential, suggesting that they are amenable to genome-wide removal. Second, our inability to replace CUU with CUG, UUG, and GUG at position 162 in *rplQ* demonstrates that synonymous codons can be non-equivalent in unpredictable ways. Nevertheless, our ability to successfully remove all instances of 13 codons from a panel of highly expressed essential genes indicates that radical genome recoding is feasible. Finally, most of the recoded genes displayed reduced fitness, and combining the current designs into a single genome could lead to unacceptable fitness impairment. In contrast, we did not observe significantly altered growth rates for the CoS-MAGE strains in which only forbidden codons were changed (Table S6-5). Therefore, our future strategies for genome-wide codon reassignment will only change codons of interest while selecting for variants with normal growth. This approach leverages diversity and evolution to

overcome such uncharacterized genome design constraints, allowing researchers to focus on creating genomes possessing new and useful functions.

Materials and Methods

All DNA oligonucleotides (Table S6-8) were purchased with standard purification and desalting from Integrated DNA Technologies. The Oligo Library Synthesis (OLS) array used for synthesizing radically recoded genes was generated on a DNA microchip, processed, and delivered as a ~1-10 pmol lyophilized pool of oligos by Agilent Technologies (Carlsbad, CA).

Cultures were grown at 34 °C with aeration in LB-Lennox (LB^L; 10 g/L Bacto tryptone, 5 g/L sodium chloride, 5 g/L yeast extract) and colonies were grown on LB^L-agar plates (LB^L with 15 g/L Bacto agar). LB^L media was supplemented with one or more of the following selective agents: carbenicillin (50 μg/mL), sodium dodecyl sulfate (SDS; 0.005% w/v), kanamycin (30 μg/mL). Colicin E1 was obtained via expression in strain JC411 (*31*), and purified as previously described.

NAT_*kan*^{R} cassette preparation: Kanamycin resistance (*kan*^{R}) cassettes were inserted *via* λ Red recombination (*26, 32*) downstream of essential ribosomal genes, in order to test whether polar effects from inserting *kan*^{R} impair fitness. These "NAT_*kan*^{R}" cassettes were PCR amplified using primers that introduced 50 bp of genomic homology on either side of the intended *kan*^{R} insertion site (Kapa HiFi Ready Mix; manufacturer's protocols). PCR products were SPRI purified as previously described (*33*), eluted in deionized water (dH_2O), and checked on a 1% agarose gel for correct size and purity before being recombined as described below.

Recoded gene cassette preparation: Recoded essential genes (Table S6-9) were synthesized from an Agilent OLS array as previously described (*24*). Due to their size, the *prfB*

and *rpsA* genes were difficult to synthesize in one piece, so they were each synthesized in two pieces, which were then assembled *via* isothermal assembly (*34*). All synthesized recoded cassettes were fused to a downstream kanamycin resistance gene (kan^R) *via* isothermal assembly (*34*). The crude isothermal assemblies were PCR amplified using primers (Table S6-8) that introduced 50 bp of genomic homology on either side of the recoded gene and kan^R (Kapa HiFi Ready Mix; manufacturer's protocols). Full-length cassettes were SPRI purified as previously described (*33*), eluted in dH_2O, and checked on a 1% agarose gel for correct size and purity before being recombined as described below.

Partially recoded cassette preparation: Partially recoded gene cassettes were prepared using the full-length recoded gene cassettes (described above) as template (Kapa HiFi Ready Mix; manufacturer's protocols). While the same reverse primers were used, new forward primers were designed to hybridize inside the recoded cassette and to introduce 50 bp homology regions matching the natural sequence, so that only the C-terminal portion of the gene would be recoded (Figure 6-1B).

We prepared two types of partially recoded cassettes. The less stringent version recoded exactly half of the gene. The more stringent version recoded all except for the first 30 codons of the gene. Partially recoded cassettes were SPRI purified as previously described (*33*), eluted in dH_2O, and checked on a 1% agarose gel for correct size and purity before being recombined as described below.

CoS-MAGE selectable marker preparation: To maximize the number of alleles that could simultaneously be replaced per recombinant, we used Co-Selection Multiplex Automated Genome Engineering (CoS-MAGE) with *tolC* or *bla* as co-selectable markers (*18, 27*). In most cases, 90 nt MAGE oligos were designed to replace several forbidden codons. We performed

CoS-MAGE in an EcNR2.*xseA*$^{-}$ background, which has ExoVII inactivated in order to minimize allele loss near the 3' end of the MAGE oligos (*35*). Since the ribosomal genes are clustered in different regions of the genome, selectable markers needed to be placed in multiple different genomic locations in order to provide co-selection in adequate proximity (~500 kb) to the target ribosomal genes. Therefore, we prepared two *tolC* cassettes (*tolC*.3502900 for *rpsL*, *rplQ*, *rplO*, *rpsG*, *rplD*, *rpsD*, *rpsC*, and *rplB*; *tolC*.4427600 for *rpsR*, *rplL*, and *rplJ*) using Kapa HiFi Ready Mix (manufacturer's protocols) and PCR primers that introduced 50 bp of flanking genomic homology (Table S6-8). The *tolC* cassettes were purified using Qiagen's PCR purification kit (manufacturer's protocols, eluted in dH_2O) before being recombined as described in the "gene and allele replacement" methods section. For *rpsA* co-selection, *bla* was already present in the λ prophage of EcNR2.

Gene and allele replacement: All CoS-MAGE oligonucleotides and Nat_*Kan*^{R}, fully recoded, and partially recoded cassettes (described above) were recombined into EcNR2 (*E. coli* MG1655 Δ*mutS*::*cat* Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*]) as previously described (*18*). Briefly, EcNR2 was grown to mid-log phase (OD_{600} between 0.4 and 0.6), induced to express λ Red for 15 minutes in a 42 °C shaking water bath, and chilled on ice. For each recombination, 1 mL of induced culture was washed twice in 1 mL cold dH_2O, and then the cell pellet was resuspended in 50 µL of dH_2O containing the DNA to be recombined. For PCR products, 1-2 ng/µL was used; to inactivate selectable markers for CoS-MAGE, a 90mer oligonucleotide was used at a final concentration of 1 µM; for CoS-MAGE, 90mer oligonucleotides were pooled at a final concentration of ≤ 5 µM. A BioRad GenePulser™ was used for electroporation (0.1 cm cuvette, 1.78 kV, 200 Ω, 25 µF), and electroporated cells were

allowed to recover in 3 ml LB^L in a rotator drum at 34°C for at least 3 hours before plating on appropriate selective media.

Recombinant clones were selected on LB^L-agar supplemented with kanamycin, and then re-streaked on fresh LB^L-agar supplemented with kanamycin to ensure monoclonality. Monoclonal colonies were then grown in a 96-well format (150 μL LB^L supplemented with kanamycin) in preparation for genetic analysis.

To prepare the EcNR2.*xseA*$^-$ strains for CoS-MAGE, we deleted the endogenous *tolC* from the genome using the tolC.90.del oligo and selected for recombinants *via* Colicin E1 selection (*18*). We then separately introduced the *tolC* co-selection cassettes (one per CoS-MAGE strain) and selected on LB^L supplemented with SDS. Finally, we inactivated *tolC* by introducing a nonsense mutation and a frameshift using the tolC-r_null_mut* oligo. For *bla* co-selection, we used the bla_mut* oligo to inactivate *bla* (present in the λ prophage) and screened for carbenicillin-sensitive recombinants by replica plating on LB^L supplemented with carbenicillin.

CoS-MAGE: CoS-MAGE was performed as previously described (*27*), using 0.5 μM of each MAGE oligo and 0.5 μM of the appropriate co-selection oligo to revert *tolC*.3502900 (*rpsL*, *rplQ*, *rplO*, *rpsG*, *rplD*, *rpsD*, *rpsC*, *rplB*), *tolC*.4427600 (*rpsR*, *rplL*, *rplJ*), or *bla* (*rpsA*). MAGE (without co-selection) (*19*) was performed on *rpsP* and *rpsB* because they were distant from the available co-selectable markers and only had 4 codons to be removed. CoS-MAGE recombinants were selected on LB^L-agar supplemented with SDS (for *tolC*) or LB^L-agar supplemented with carbenicillin (for *bla*), and MAGE recombinants were grown on LB^L-agar without selection. Monoclonal colonies were picked into a 96-well plate and grown under the appropriate selection at 34 °C with shaking.

Recombinant clone genotyping: Recombinant clones were first screened by PCR, then validated by Sanger sequencing. For the fully recoded genes, we performed 3 PCR reactions for each clone. As diagramed in Figure 6-1B, the three sets of primers hybridized to the natural gene sequence (NAT), the recoded gene sequence (SYN), and the flanking genomic region (BND). PCR reactions (10 µL each) were performed with Kapa 2G Fast Ready Mix according to the manufacturer's protocols. Adequate primer specificity was observed with a 58 °C annealing temperature. Desired recombinants had no NAT amplicon, a gene-sized SYN amplicon, and a BND amplicon 847 bp larger than that of the wild type negative control. Partially (C-terminally) recoded recombinants were screened using the NAT forward and SYN reverse primers (desired recombinants had a gene-sized amplicon) and BND primers (desired recombinants showed an 847 bp increase in amplicon size). All putative recombinants that passed the PCR assay were Sanger sequenced (Genewiz or Eton Bioscience Inc.) using the forward BND primers and/or kanR.seqOUT-Nr2.

CoS-MAGE recombinants were typically sequenced without initial Multiplex Allele Specific Colony PCR (MASC-PCR (*18*)) screening because the targeted alleles were too close together to allow for the amplification of discrete bands. However, well-separated alleles were screened via MASC-PCR with standard protocols (*18*) prior to Sanger sequencing validation.

Doubling time analysis: Doubling times (Figure 6-2, Tables S6-4 to S6-5) were determined for all recoded clones using LB^L and Teknova HiDef Azure media. Kinetic growth curves were performed in triplicate on a Biotek H4 plate reader with OD_{600} measurements at 5 minute intervals. Cultures were grown in a flat-bottom 96-well plate (in 150 µL of LB^L supplemented with carbenicillin) with shaking at 34 °C. Doubling times were determined by $t_{double} = c*\ln(2)/m$, where c = 5 minutes per time point and m is the maximum slope of $\ln(OD_{600})$

smoothed across 5 contiguous time points (20 minutes). We typically calculate doubling time in this manner so as to accommodate strains that achieve lower maximum optical densities. Each data point in Figure 6-2 represents the average doubling time of an individual strain with one ribosomal gene partially or fully recoded (n = 3). Each replicate was prepared by passaging from the previous one. All strains are based on EcNR2 or EcNR2.xseA^{-} [doubling times under assay conditions for these strains are 49 +/- 4 minutes in LB^{L} and 84 +/- 5 minutes in Teknova HiDef Azure Media (12 replicates per condition)].

Supplemental material

Supplemental material for CHAPTER 6 can be found in APPENDIX E or at <http://www.sciencemag.org/content/suppl/2013/10/16/342.6156.361.DC1/Lajoie2.SM.pdf>.

References

1. A. Ambrogelly, S. Palioura, D. Soll, Natural expansion of the genetic code. *Nat Chem Biol* **3**, 29 (2007).

2. M. G. Schafer *et al.*, The Establishment of Genetically Engineered Canola Populations in the U.S. *PLoS One* **6**, e25736 (2011).

3. M. J. Lajoie *et al.*, Genomically Recoded Organisms Expand Biological Functions. *Science* **342**, 357 (Oct 18, 2013).

4. C. C. Liu, P. G. Schultz, Adding New Chemistries to the Genetic Code. *An. Rev. Biochem.* **79**, 413 (2010).

5. M. Frenkel-Morgenstern *et al.*, Genes adopt non-optimal codon usage to generate cell cycle-dependent oscillations in protein levels. *Mol. Syst. Biol.* **8**, (2012).

6. N. T. Ingolia, S. Ghaemmaghami, J. R. S. Newman, J. S. Weissman, Genome-Wide Analysis in Vivo of Translation with Nucleotide Resolution Using Ribosome Profiling. *Science* **324**, 218 (Apr, 2009).

7. T. Tuller *et al.*, An evolutionarily conserved mechanism for controlling the efficiency of protein translation. *Cell* **141**, 344 (Apr, 2010).

8. H. Gingold, Y. Pilpel, Determinants of translation efficiency and accuracy. *Mol. Syst. Biol.* **7**, (Apr, 2011).

9. J. B. Plotkin, G. Kudla, Synonymous but not the same: the causes and consequences of codon bias. *Nat. Rev. Genet.* **12**, 32 (Jan, 2011).

10. H. Akashi, Synonymous codon usage in Drosophila melanogaster: Natural selection and translational accuracy. *Genetics* **136**, 927 (Mar, 1994).

11. E. Angov, Codon usage: Nature's roadmap to expression and folding of proteins. *Biotechnol. J.* **6**, 650 (2011).

12. S. Pechmann, J. Frydman, Evolutionary conservation of codon optimality reveals hidden signatures of cotranslational folding. *Nature structural & molecular biology* **20**, 237 (2013-Feb, 2013).

13. M. Bulmer, The selection-mutation-drift theory of synonymous codon usage. *Genetics* **129**, 897 (November 1, 1991, 1991).

14. A. Shinhara *et al.*, Deep sequencing reveals as-yet-undiscovered small RNAs in Escherichia coli. *BMC Genomics* **12**, (2011).

15. G.-W. Li, E. Oh, J. S. Weissman, The anti-Shine-Dalgarno sequence drives translational pausing and codon choice in bacteria. *Nature* **484**, 538 (2012).

16. K. Nevo-Dinur, A. Nussbaum-Shochat, S. Ben-Yehuda, O. Amster-Choder, Translation-Independent Localization of mRNA in E. coli. *Science* **331**, 1081 (Feb, 2011).

17. D. G. Gibson *et al.*, Creation of a Bacterial Cell Controlled by a Chemically Synthesized Genome. *Science* **329**, 52 (Jul, 2010).

18. F. J. Isaacs *et al.*, Precise Manipulation of Chromosomes in Vivo Enables Genome-Wide Codon Replacement. *Science* **333**, 348 (Jul, 2011).

19. H. H. Wang *et al.*, Programming cells by multiplex genome engineering and accelerated evolution. *Nature* **460**, 894 (Aug, 2009).

20. Y. Yamazaki, Niki, H., & Kato, J., Profiling of Escherichia coli Chromosome database. *Methods Mol. Biol.* **416**, 385 (2008).

21. K. Higashi *et al.*, Enhancement of +1 Frameshift by Polyamines during Translation of Polypeptide Release Factor 2 in Escherichia coli. *J. Biol. Chem.* **281**, 9527 (April 7, 2006, 2006).

22. M. Kaczanowska, M. Ryden-Aulin, Ribosome biogenesis and the translation process in Escheticha coli. *Microbiology and Molecular Biology Reviews* **71**, 477 (Sep, 2007).

23. M. T. Sykes, E. Sperling, S. S. Chen, J. R. Williamson, Quantitation of the Ribosomal Protein Autoregulatory Network Using Mass Spectrometry. *Analytical Chemistry* **82**, 5038 (Jun, 2010).

24. S. Kosuri *et al.*, Scalable gene synthesis by selective amplification of DNA pools from high-fidelity microchips. *Nat Biotech* **28**, 1295 (2010).

25. D. G. Gibson *et al.*, Enzymatic assembly of DNA molecules up to several hundred kilobases. *Nat Meth* **6**, 343 (2009).

26. D. G. Yu *et al.*, An efficient recombination system for chromosome engineering in Escherichia coli. *Proc. Natl. Acad. Sci. U. S. A.* **97**, 5978 (May, 2000).

27. P. A. Carr *et al.*, Enhanced multiplex genome engineering through co-operative oligonucleotide co-selection. *NAR*, 1 (2012).

28. R. Zell, H. J. Fritz, DNA mismatch repair in Escherichia coli counteracting the hydrolytic deamination of 5-methyl-cytosine residues. *Embo J.* **6**, 1809 (Jun, 1987).

29. M. A. Sørensen *et al.*, Over Expression of a tRNALeu Isoacceptor Changes Charging Pattern of Leucine tRNAs and Reveals New Codon Reading. *Journal of Molecular Biology* **354**, 16 (2005).

30. A. C. Forster, G. M. Church, Towards synthesis of a minimal cell. *Mol Syst Biol* **2**, (2006).

CHAPTER 7

Conclusions and Future Projects

Acknowledgements:
D. Mandell, C. Gregg, M. Napolitano, M. Landon, G. Kuznetsov, D. Goodman, M. Güell, J. Norville, W. Chew, J. Mosberg, J. Rogers, S. Yaung, M. Moosburner, M. Chung, X. Rios, J. Cruz, and L. Govindarajan are collaborating on additional projects that have not yet been published. Funding was from Department of Energy [DE-FG02-02ER63445], NSF [SA5283-11210], DARPA [N66001-12-C-4040], and Department of Defense NDSEG Fellowship (M.J.L.).

Genome design is the major challenge for engineering genomes possessing new and useful properties. How do you fundamentally change the properties of a genome without introducing deleterious design flaws? My early graduate work focused on developing efficient *in vivo* genome engineering technologies (*1-5*) capable of extensively engineering the *Escherichia coli* genome. Important features of these technologies are: (1) *in vivo* editing allows real time assessment of intended and unintended genome alterations, reducing the risk that one synthesis or design flaw will cause an entire genome to fail, (2) exploiting evolution to remove deleterious alleles from the population makes it possible to test aggressive designs containing potentially deleterious alleles, (2) multiplex allele replacement facilitates rapid accumulation of desired changes (3), and parallelization makes the unit of engineering smaller (*i.e.*, genome segments vs. whole genomes), facilitating efficient genome assembly and rapid mapping of design flaws.

For my main project, I used these technologies to replace all instances of the UAG stop codon with the synonymous UAA codon in *E. coli* (*2, 6*). With this strain in hand, I tested our hypotheses that genomically recoded organisms (GROs) would improve NSAA incorporation and resist viruses. Although the results of these tests were promising, it was clear that additional codons must be reassigned in order to fully realize our goals to expand the genetic code, to block virus infection, and to mistranslate genetic material transferred to/from natural organisms. However, reassigning additional codons was technologically and biologically a daunting task. The next-rarest codons in *E. coli* are 10-fold more abundant than UAG, and there is extensive evidence that sense codons perform additional functions beyond choosing an amino acid (see Chapter 6). We wondered whether we could find instances of codons that were essential and could not be changed to any other codon, so we performed a pilot experiment in which we removed 13 rare codons from 42 highly expressed essential genes. Although we were successful

in finding many genome designs that were not viable, we did not find any codons that could not be changed (*7*), suggesting that it may be possible to recode some of these codons genome-wide.

Extrapolating from the first study in which we completely reassigned the UAG stop codon and the second study in which we identified 13 codons that could be potentially reassigned, we are currently attempting to engineer a genome with 7 codons reassigned, requiring 62,733 codon changes (Figure 7-1). To accomplish this, we are developing an efficient pipeline based on computer-aided genome design, chip-based DNA synthesis, *in vitro* DNA fragment assembly, and *in vivo* genome assembly to synthesize a recoded genome. Even more so than for our previous genetic code engineering projects, there is a considerable risk for design flaws, so our pipeline is designed to facilitate rapid troubleshooting. This includes crude mapping of design flaws using recombineering, fine mapping/troubleshooting of design flaws using MAGE, and efficient genome analysis software to track the effects of intended and unintended mutations.

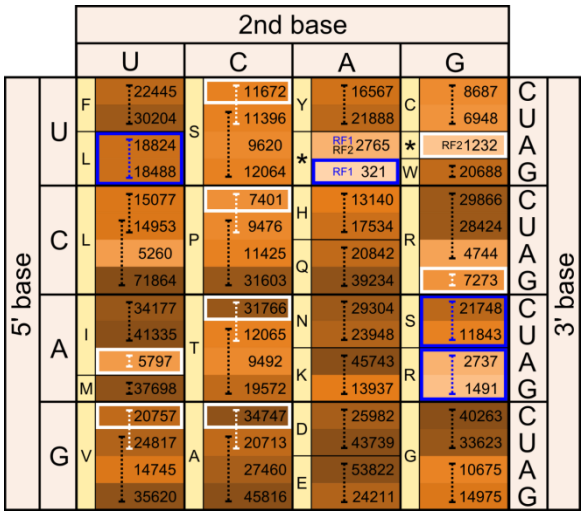


Figure 7-1. Proposed future genetic code reassignment. *E. coli* MG1655 codon usage heat map; brightness increases as codon usage decreases. Black numbers are total codon usage based on NC_000913.2 (National Center for Biotechnology Information, 1 September 2011). The anticodon–codon pairings (*29, 30*) are illustrated as dashed brackets; blue indicates anticodons that are targeted for eventual removal; white indicates additional anticodons that could be potentially removed. Blue boxes denote the 7 codons that we intend to remove from the genome prior to reassignment (62,733 codon changes required). White boxes denote additional codons that could be removed (164,505 codon changes required). Amino acids are indicated in the yellow side bars.

In addition to producing a powerful chassis organism for biotechnology, it is important to ensure safety. Although it is unlikely that GROs would survive well in the environment (even if they are resistant to viruses), safeguards can be implemented to ensure their safety. As mentioned

above, in the case of accidental horizontal gene transfer, differences in the genetic code would genetically isolate the recombinant DNA from functioning in natural organisms. Additionally, we are engineering C321.ΔA to be metabolically dependent on NSAAs for survival, ensuring that this strain cannot survive in the environment. We are accomplishing this by redesigning essential proteins so that they are dependent on the correct incorporation of a NSAA for proper translation, folding, and function. Dependence on unnatural compounds that are not produced in any known natural environment would provide a more robust alternative to traditional auxotrophies that can be overcome in metabolically rich environments.

We have a number of additional projects underway:

- Fixing C321.ΔA: Although we have begun work on the second generation of GROs, strain C321.ΔA (*6*) is a useful resource for studying the genetic code and expanding protein functions by incorporating unique NSAAs into proteins. Therefore, it would be beneficial to improve this strain as a resource for the scientific community. We have analyzed the C321.ΔA genome to identify the cause of its reduced fitness. Of the 355 off-target mutations, we identified a subset of ~100 candidates that are most likely to be deleterious, and we have combinatorially reverted them using MAGE. After ~30 cycles of MAGE, we improved the doubling time from ~80 minutes/doubling to ~60 minutes/doubling (the ancestral strain has ~50 minutes/doubling under our assay conditions). We are now analyzing the mutations in these improved strains in order to identify the causative alleles. We will implement these fixes in C321.ΔA and provide the improved strain as a resource for the scientific community.
- AGR recoding: We are engineering an *E. coli* strain in which all 123 AGR codons in its essential genes are changed to alternative codons. This project will (1) pressure-test the

AGR codons that are expected to be most recalcitrant for recoding, (2) provide actionable sequence information for a synthetic genome with reassigned AGR codons, and (3) illuminate design constraints that will help improve future genome designs.

- Extending MAGE to other organisms: We have developed powerful technologies for engineering *Escherichia coli* genomes, but we would like to engineer other organisms, too. While the bacteriophage λ Red recombination system mediates extremely efficient genome engineering in *E. coli*, it does not extend well to disparate organisms. Conveniently, homologs of this system exist across many organisms (*8*), and there has been some success reported for expanding recombineering to other organisms (*9-13*). We have developed a general method to select efficient recombineering activity from a panel of metagenomic recombinases in any organism of interest. We piloted this method in *E. coli* and identified additional recombinases that yielded similar recombination frequencies to λ Red. We are now applying this method to useful organisms such as *P. aeruginosa*, *S. aureus*, *A. bayli*, and cyanobacteria.
- Improving *tolC*: In continued effort to improve technologies for efficient *in vivo* genome engineering, we rationally engineered an *E. coli* strain with improved robustness of the *tolC* dual selectable marker (Gregg, C.J. and Lajoie, M.J. *in revision*). Previously, the negative selection exhibited a failure rate that was too high for continued use without intermittent screening for desired recombinants. The improved strain is capable of continuous CoS-MAGE cycling, which has made it significantly easier to remove AGR codons from essential *E. coli* genes.

References

1. J. A. Mosberg, M. J. Lajoie, G. M. Church, Lambda red recombineering in escherichia coli occurs through a fully single-stranded intermediate. *Genetics* **186**, 791 (Nov, 2010).

2. F. J. Isaacs *et al.*, Precise manipulation of chromosomes in vivo enables genome-wide codon replacement. *Science* **333**, 348 (Jul, 2011).

3. J. A. Mosberg, C. J. Gregg, M. J. Lajoie, H. H. Wang, G. M. Church, Improving lambda red genome engineering in *Escherichia coli* via rational removal of endogenous nucleases. *PLoS One* **7**, e44638 (2012).

4. P. A. Carr *et al.*, Enhanced multiplex genome engineering through co-operative oligonucleotide co-selection. *Nucleic Acids Res.*, (May 25, 2012, 2012).

5. M. J. Lajoie, C. J. Gregg, J. A. Mosberg, G. C. Washington, G. M. Church, Manipulating replisome dynamics to enhance lambda Red-mediated multiplex genome engineering. *Nucleic Acids Res.* **40**, e170 (2012-Dec-1, 2012).

6. M. J. Lajoie *et al.*, Genomically recoded organisms expand biological functions. *Science* **342**, 357 (Oct 18, 2013).

7. M. J. Lajoie *et al.*, Probing the limits of genetic recoding in essential genes. *Science* **342**, 361 (Oct 18, 2013).

8. S. Datta, N. Costantino, X. Zhou, D. L. Court, Identification and analysis of recombineering functions from Gram-negative and Gram-positive bacteria and their phages. *Proceedings of the National Academy of Sciences* **105**, 1626 (February 5, 2008, 2008).

9. J. C. van Kessel, G. F. Hatfull, Recombineering in Mycobacterium tuberculosis. *Nat. Methods* **4**, 147 (Feb, 2007).

10. R. G. Gerlach, D. Jackel, S. U. Holzer, M. Hensel, Rapid Oligonucleotide-Based Recombineering of the Chromosome of Salmonella enterica. *Applied and Environmental Microbiology* **75**, 1575 (Mar, 2009).

11. B. Swingle, Z. M. Bao, E. Markel, A. Chambers, S. Cartinhour, Recombineering Using RecTE from Pseudomonas syringae. *Applied and Environmental Microbiology* **76**, 4960 (Aug, 2010).

12. X. Rios *et al.*, Stable Gene Targeting in Human Cells Using Single-Strand Oligonucleotides with Modified Bases. *PLoS One* **7**, e36697 (2012).

13. J. P. van Pijkeren, R. A. Britton, High efficiency recombineering in lactic acid bacteria. *Nucleic Acids Res.* **40**, (May, 2012).

APPENDIX A

Supplemental Material for Lambda Red Recombination in Escherichia coli Occurs Through a Fully Single-Stranded Intermediate

This supplemental material is reproduced with permission from its initial publication:

Mosberg JA[*], **Lajoie MJ[*]**, Church GM (2010) *Lambda Red Recombination in Escherichia coli Occurs Through a Fully Single-Stranded Intermediate***. GENETICS:** Vol. 186, 791-799

Tables and Figures have been renamed to be consistent with CHAPTER 2.

FILE S1: Supplemental Materials and Methods

Preparation of DNA constructs: PCR primers were ordered from Integrated DNA Technologies and are listed and described in Table S2-1. All primers were ordered with standard desalting, except dual-biotinylated primers, which were HPLC-purified. Antibiotic resistance insertion cassettes were generated using long PCR primers containing 45 bp genome homology regions on the 5′ end, followed by roughly 20 bp of homology to the antibiotic resistance gene to be amplified. The insertion cassettes were designed such that the resistance gene was inserted 46 base pairs into the coding DNA sequence in the case of *lacZ*, and directly after the start codon in the cases of *malK* and *tolC*. This set of PCRs was performed using Qiagen HotStarTaq Plus Master Mix. Final primer concentrations were 0.4 μM, and templates were resuspended bacterial colonies bearing the desired resistance gene (the tn*903 aphA1* gene for kanamycin resistance, the *Sh ble* gene for zeocin resistance, and the tn*21 aadA1* gene for spectinomycin resistance - each cassette contained promoter and terminator sequences flanking the resistance gene). These PCRs were heat activated at 95 °C for 6:00, and then cycled 30 times using a denaturation step of 94 °C for 0:30, an annealing step of 56 °C for 0:30, and an extension step of 72 °C for 2:30. After a final 5:00 extension step at 72 °C, PCRs were held at 4 °C, then purified via 1% agarose gel extraction using the Qiagen gel extraction kit. DNA samples were quantitated using a NanoDrop™ ND1000 spectrophotometer.

These constructs were used as template for subsequent PCRs to generate dual-biotinylated dsDNA constructs. In each reaction, one primer contained a 5′ dual-biotin tag. The other primer was unmodified, or contained four 5′ phosphorothioate bonds. Phosphorothioate bonds were used in the experiment comparing leading-targeting and lagging-targeting ssDNA, with the rationale that this would increase recombination frequency by mitigating exonuclease degradation. PCR conditions were as above, but with 1 μM primers, a 1:30 extension step, and 0.1 ng of the relevant insertion construct used as template. PCR products were purified using the Qiagen PCR purification kit.

These dual-biotinylated dsDNA constructs were used to generate ssDNA via a biotin capture protocol. In this method, the dual-biotinylated DNA strand is bound by streptavidin-coated magnetic beads. Next, the dsDNA is chemically melted, allowing the non-biotinylated strand to be collected from the supernatant, while the biotinylated strand is retained by the beads. Invitrogen DynaBeads® MyOne™ Streptavidin C1 beads were washed twice with 2x Bind and Wash buffer (10 mM Tris, 1 mM EDTA, 2 M NaCl, pH 7.5), then incubated in one initial bead volume of 1x Bind and Wash buffer, with 5 μg of dual-biotinylated dsDNA for every 100 μL of beads. This was placed on a tube rotator at room temperature for 20 minutes, after which the beads were washed twice with 1x Bind and Wash buffer. Single-stranded DNA was then released via incubation with one initial bead volume of chilled 0.125 M NaOH. Beads were vortexed for 30 seconds, incubated for 30 seconds, then placed on a magnet so that the supernatant could be collected. This process was repeated, and then the NaOH supernatants were cleaned using the Qiagen PCR purification kit. The standard PCR purification protocol was used, neutralizing the solution with a 3 M solution of pH 5.0 NaOAc, and adding an additional rinse with Buffer PE. The purity of the resulting ssDNA was confirmed by PAGE. To this end, 10 ng of purified ssDNA was loaded onto a 6% TBE non-denaturing PAGE gel (Invitrogen) and post-stained with SYBR® Gold (Invitrogen).

A similar strategy was employed for creating the internally mismatched lacZ::kanR dsDNA cassette. Two dual-biotinylated dsDNA constructs were generated, each intended to give rise to one of the two strands of the final mismatched construct. These constructs were generated in an analogous manner to those described above, with mutations arising from the PCR primers as described in Table S2-1. These dual-biotinylated dsDNA constructs were used to produce ssDNA in the same manner as above. The dual-biotin tags were arranged such that complementary strands were purified from the two constructs, so that they could be annealed together in order to form the dsDNA construct diagrammed in Figure 2-4. Purified strands were annealed in equimolar amounts (25 nM) in 5 mM Tris, 0.25 M NaCl, pH 8.0. Samples were annealed by heating to 95 °C, and then cooling the samples in a thermocycler. The temperature was decreased by 1 °C every two minutes to a final temperature of 25 °C. The resulting annealed dsDNA was purified from a 1% agarose gel using the Qiagen gel extraction kit.

In order to generate phosphorothioated variants of the lacZ::kanR mismatched dsDNA construct, the two lacZ::kanR dsDNA constructs containing the designed mutations were each amplified as above, this time using phosphorothioated primers opposite the dual-biotinylated primers. The resulting dsDNA constructs were used to purify phosphorothioated strands of ssDNA as above. The phosphorothioated ssDNA and the previously produced unmodified ssDNA strands were then each used in annealing reactions, set up so as to generate all four combinations of dsDNA (as given in Figure 2-5). Annealing and purification was carried out as described above.

Analysis of mismatched dsDNA recombinants by MAMA PCR: The mismatch amplification mutation assay (MAMA) PCR method was used to analyze the genotypes of mismatched lacZ::kanR dsDNA recombinants (Qiang et al. 2002). We used 2-bp mismatches in our mismatched lacZ::kanR cassette in order to increase the specificity of our MAMA primers and to decrease the chances of spontaneous point mutations confounding our results. We designed four primers for each mismatch locus (Figure 2-4): a forward primer corresponding to the strand 1 allele, a forward primer corresponding to the strand 2 allele, a forward primer corresponding to the wild type allele, and a universal reverse primer. Primers were designed with the 2-bp mismatches at the 3' ends so that amplification would only occur when these two nucleotides matched the recombinant colony's genotype. The L1 amplicon was 799 bp, L2 was 591 bp, L3 was 372 bp, and L4 was 205 bp. Primers were designed with a target T_m of 62 °C, and a gradient PCR (annealing temperature between 62 °C and 68 °C) determined that the optimal annealing temperature for maximum specificity and yield was 64° C. MAMA PCR reactions for loci 1&3 and loci 2&4 were each performed in a single mixture so as to minimize the number of necessary reactions. Each kanR colony was interrogated using 4 MAMA PCR reactions: strand 1 L1&L3, strand 1 L2&L4, strand 2 L1&L3, and strand 2 L2&L4. For convenience, both strand 1 reactions and both strand 2 reactions were pooled prior to agarose gel analysis. PCR template was prepared by growing a monoclonal kanR colony to stationary phase and performing a 1/100 dilution of this culture into PCR-grade water. Our 20 μL MAMA PCR reactions consisted of 10 μL Qiagen multiplex PCR master mix, 5 μL PCR grade water, 4 μL primer mix (1 μM each), and 1 μL template. PCRs were heat activated at 95 °C for 15:00, and then cycled 27 times using a denaturation step of 94 °C for 0:30, an annealing step of 64 °C for 0:30, and an extension step of 72 °C for 1:20. After a final 5:00 extension step at 72 °C, PCRs were held at 4 °C until they were analyzed on a 1.5% agarose gel stained using ethidium bromide. Strand 1 reactions for a given recombinant were loaded adjacent to corresponding strand 2 reactions for easy visual comparison. All 48 recombinants from replicate 1 were screened using wild type MAMA PCR reactions, performed in an analogous manner as above. This experiment verified that all sites that were not detected as mutants were wild type alleles. The accuracy of the MAMA PCR method was also verified by Sanger sequencing all four mismatch loci in eight recombinant colonies.

FILE S2: Mistargeted Recombinants in ssDNA Recombination

A significant number of mistargeted recombinants (antibiotic-resistant colonies that retained LacZ function) were observed for both single-stranded *lacZ::specR* cassettes in the experiments tracking strand bias in ssDNA recombination (Table S2-2). Such mistargeting also occurred with *lacZ::specR* dsDNA, but only rarely with the other *lacZ*-targeting cassettes. These recombinants likely arise when microhomology sequences within the *specR* gene anneal to regions of the *E. coli* chromosome other than *lacZ*. The observed strand bias for mistargeting may be due to differing secondary structure between the two strands. Alternatively, a leading strand bias may be observed for mistargeting, since mistargeted annealing to regions on the lagging strand could outcompete correctly targeted annealing to the less accessible leading strand. Mistargeted ($LacZ^+$) colonies were not scored as recombinants, and do not affect the broader interpretation of our results.

TABLE S2-1. Primer Sequences Used in this Study

Name	Use	Sequence	Versions used[a]
LacZ::KanR.full-f	Forward strand for generation of the initial LacZ::KanR construct	TGACCATGATTACGGATTCACTGG CCGTCGTTTTACAACGTCGTGCCTG TGACGGAAGATCACTTCG	Unmodified
LacZ::KanR.full-r	Reverse strand for generation of the initial LacZ::KanR construct	GTGCTGCAAGGCGATTAAGTTGGG TAACGCCAGGGTTTTCCCAGTAAC CAGCAATAGACATAAGCGG	Unmodified
MalK::KanR.full-f	Forward strand for generation of the initial MalK::KanR construct	AATGTTGCTGTCGATGACAGGTTG TTACAAAGGGAGAAGGGCATGCCT GTGACGGAAGATCACTTCG	Unmodified
MalK::KanR.full-r	Reverse strand for generation of the initial MalK::KanR construct	GACCTCGCCCCAGGCTTTCGTTAC ATTTTGCAGCTGTACGCTCGCAAC CAGCAATAGACATAAGCGG	Unmodified
TolC::KanR.full-f	Forward strand for generation of the initial TolC::KanR construct	AGTTTGATCGCGCTAAATACTGCT TCACCACAAGGAATGCAAATGCCT GTGACGGAAGATCACTTCG	Unmodified
TolC::KanR.full-r	Reverse strand for generation of the initial TolC::KanR construct	GAACCCAGAAAGGCTCAGGCCGAT AAGAATGGGGAGCAATTTCTTAAC CAGCAATAGACATAAGCGG	Unmodified
LacZ::ZeoR.full-f	Forward strand for generation of the initial LacZ::ZeoR construct	TGACCATGATTACGGATTCACTGG CCGTCGTTTTACAACGTCGTGGGT GTTGACAATTAATCATCGGC	Unmodified
LacZ::ZeoR.full-r	Reverse strand for generation of the initial LacZ::ZeoR construct	GTGCTGCAAGGCGATTAAGTTGGG TAACGCCAGGGTTTTCCCAGTAGC TTGCAAATTAAAGCCTTCG	Unmodified
LacZ::SpecR.full-f	Forward strand for generation of the initial LacZ::SpecR construct	TGACCATGATTACGGATTCACTGG CCGTCGTTTTACAACGTCGTGCAG CCAGGACAGAAATGC	Unmodified
LacZ::SpecR.full-r	Reverse strand for generation of the initial LacZ::SpecR construct	GTGCTGCAAGGCGATTAAGTTGGG TAACGCCAGGGTTTTCCCAGTTGC AGAAATAAAAAGGCCTGC	Unmodified
LacZ.short-f	Forward strand for generation of all dual-biotinylated LacZ-targeting constructs	TGACCATGATTACGGATTCACT	Unmodified, Phosphorothioated, Dual-biotinylated
LacZ.short-r	Reverse strand for generation of all dual-biotinylated LacZ-targeting constructs	GTGCTGCAAGGCGATTAA	Phosphorothioated, Dual-biotinylated

TABLE S2-1 (Continued).

MalK::KanR.short-f	Forward strand for generation of dual-biotinylated MalK::KanR constructs	AATGTTGCTGTCGATGACAGG	Phosphorothioated, Dual-biotinylated
MalK::KanR.short-r	Reverse strand for generation of dual-biotinylated MalK::KanR constructs	GACCTCGCCCCAGGC	Phosphorothioated, Dual-biotinylated
TolC::KanR.short-f	Forward strand for generation of dual-biotinylated TolC::KanR constructs	AGTTTGATCGCGCTAAATACTG	Phosphorothioated, Dual-biotinylated
TolC::KanR.short-r	Reverse strand for generation of dual-biotinylated TolC::KanR constructs	GAACCCAGAAAGGCTCAGG	Phosphorothioated, Dual-biotinylated
MM.LacZ::KanR.AA-f	Forward strand for generation of Construct AA (For the creation of mismatched LacZ::KanR)	TGACCATGAAAACGGATTCACTGGCCGTCGTTAAACAACGTCGTGCCTGTGACGGAAGATCACTTCG	Unmodified
MM.LacZ::KanR.AA-r	Reverse strand for generation of Construct AA (For the creation of mismatched LacZ::KanR)	GTGCTGCAAAACGATTAAGTTGGGTAACGCCAAAGTTTTCCCAGTAACCAGCAATAGACATAAGCGG	Unmodified
MM.LacZ::KanR.CC-f	Forward strand for generation of Construct CC (For the creation of mismatched LacZ::KanR)	TGACCATGACCACGGATTCACTGGCCGTCGTTCCACAACGTCGTGCCTGTGACGGAAGATCACTTCG	Unmodified
MM.LacZ::KanR.CC-r	Reverse strand for generation of Construct CC (For the creation of mismatched LacZ::KanR)	GTGCTGCAACCCGATTAAGTTGGGTAACGCCACCGTTTTCCCAGTAACCAGCAATAGACATAAGCGG	Unmodified
MM.AA.short-f	Forward strand for generation of dual-biotinylated Construct AA	TGACCATGAAAACGGATTCAC	Unmodified, Phosphorothioated
MM.AA.short.DB-r	Reverse strand for generation of dual-biotinylated Construct AA	GTGCTGCAAAACGATTAAGTTG	Dual-biotinylated
MM.CC.short.DB-f	Forward strand for generation of dual-biotinylated Construct CC	TGACCATGACCACGGATTC	Dual-biotinylated
MM.CC.short-r	Reverse strand for generation of dual-biotinylated Construct CC	GTGCTGCAACCCGATTAAG	Unmodified, Phosphorothioated
Kan.L1.AA.set1	Forward MAMA PCR primer corresponding to the forward strand specific mismatch at position 1 in MM.LacZ::KanR	CAGGAAACAGCTATGACCATGAAA	Unmodified

TABLE S2-1 (Continued).

Kan.L2.AA.set1	Forward MAMA PCR primer corresponding to the forward strand specific mismatch at position 2 in MM.LacZ::KanR	GATTCACTGGCCGTCGTTAA	Unmodified
Kan.L3.TT.set1	Forward MAMA PCR primer corresponding to the forward strand specific mismatch at position 3 in MM.LacZ::KanR	ATTGCTGGTTACTGGGAAAACTT	Unmodified
Kan.L4.TT.set1	Forward MAMA PCR primer corresponding to the forward strand specific mismatch at position 4 in MM.LacZ::KanR	GGCGTTACCCAACTTAATCGTT	Unmodified
Kan.L1.AA.set2	Forward MAMA PCR primer corresponding to the reverse strand specific mismatch at position 1 in MM.LacZ::KanR	GGAAACAGCTATGACCATGACC	Unmodified
Kan.L2.AA.set2	Forward MAMA PCR primer corresponding to the reverse strand specific mismatch at position 2 in MM.LacZ::KanR	TCACTGGCCGTCGTTCC	Unmodified
Kan.L3.TT.set2	Forward MAMA PCR primer corresponding to the reverse strand specific mismatch at position 3 in MM.LacZ::KanR	GCTGGTTACTGGGAAAACGG	Unmodified
Kan.L4.TT.set2	Forward MAMA PCR primer corresponding to the reverse strand specific mismatch at position 4 in MM.LacZ::KanR	GCGTTACCCAACTTAATCGGG	Unmodified
Kan.L1.TT.setWT	Forward MAMA PCR primer corresponding to the wild type allele at position 1 in MM.LacZ::KanR	CAGGAAACAGCTATGACCATGATT	Unmodified
Kan.L2.TT.setWT	Forward MAMA PCR primer corresponding to the wild type allele at position 2 in MM.LacZ::KanR	GATTCACTGGCCGTCGTTTT	Unmodified
Kan.L3.CC.setWT	Forward MAMA PCR primer corresponding to the wild type allele at position 3 in MM.LacZ::KanR	GCTGGTTACTGGGAAAACCC	Unmodified

TABLE S2-1 (Continued).

Kan.L4.CC.setWT	Forward MAMA PCR primer corresponding to the wild type allele at position 4 in MM.LacZ::KanR	GCGTTACCCAACTTAATCGCC	Unmodified
Kan.L1.rev	Reverse MAMA PCR primer that is complementary to Kan.L1.AA.set1, Kan.L1.AA.set2, and Kan.L1.TT.setWT	ATGCATTTCTTTCCAGACTTGTTCA	Unmodified
Kan.L2.rev	Reverse MAMA PCR primer that is complementary to Kan.L2.AA.set1, Kan.L2.AA.set2, and Kan.L2.TT.setWT	GCATCAACAATATTTTCACCTGAA TCA	Unmodified
Kan.L3.rev	Reverse MAMA PCR primer that is complementary to Kan.L3.TT.set1, Kan.L3.TT.set2, and Kan.L3.CC.setWT	CTGTAGCCAGCTTTCATCAACA	Unmodified
Kan.L4.rev	Reverse MAMA PCR primer that is complementary to Kan.L4.TT.set1, Kan.L4.TT.set2, and Kan.L1.CC.setWT; Sequencing primer for MAMA PCR validation	AGGGGACGACGACAGTATC	Unmodified
Kan.L1.L2.seq	Sequencing primer for MAMA PCR validation	TAGCTCACTCATTAGGCACC	Unmodified

[a] Phosphorothioated primers contain four phosphorothioate linkages on the 5′ end. Dual-biotinylated primers contain a dual-biotin tag on the 5′ end.

TABLE S2-2

Mistargeting Frequencies from ssDNA Strand Bias Recombination Experiment

DNA	Avg. Lagging-Targeting ssDNA Mistargeting Frequency (Standard Dev.)	% Mistargeted Recombinants	Avg. Leading-Targeting ssDNA Mistargeting Frequency (Standard Dev.)	% Mistargeted Recombinants
lacZ::kanR	1.1E-07 (+/- 1E-08)	0.19%	5.0E-08 (+/- 1.8E-08)	0.44%
lacZ::zeoR	1.8E-08 (+/- 3.0E-08)	0.0082%	2.6E-08 (+/- 3.2E-08)	0.18%
lacZ::specR	3.3E-06 (+/- 3.2E-06)	7.0%	5.5E-06 (+/- 1.7E-06)	76%

TABLE S2-3

Full MAMA PCR Results from Mismatched *lacZ::kanR* Recombination

Mutation Inheritance Pattern[a]	Count (Replicate 1)	Count (Replicate 2)	Total	Grouped by Category[b]
1/1/1/1	0	0	0	68
1/1/1/WT	10	8	18	
WT/1/1/1	0	3	3	
WT/1/1/WT	24	23	47	
2/2/2/2	0	0	0	7
2/2/2/WT	0	1	1	
WT/2/2/2	0	4	4	
WT/2/2/WT	0	2	2	
WT/1/1/2	4	1	5	10
WT/1/2/WT	4	0	4	
WT/1/2/2	0	1	1	
2/2/1/WT	0	0	0	9
WT/2/1/WT	5	3	8	
WT/2/1/1	0	1	1	
Ambiguous	1	1	2	2
Sum	**48**	**48**	**96**	**96**

[a] Loci 1-4 are listed in order. "1" indicates inheritance from strand 1, "2" indicates inheritance from strand 2, and "WT" indicates no mutation (*i.e.* a wild type allele)

[b] Grouped based on the manner of Exo processing that is implied, as detailed in Table 2-1.

APPENDIX B

Supplemental Material for Manipulating Replisome Dynamics and DNA Exonucleases to Enhance Lambda Red-Mediated Multiplex Genome Engineering

This supplemental material is reproduced with permission from its initial publication:

Lajoie MJ*, Gregg CJ*, Mosberg JA*, Washington GC, Church GM (2012) *Manipulating Replisome Dynamics to Enhance Lambda Red-Mediated Multiplex Genome Engineering*. **NAR**; doi: 10.1093/nar/gks751

Tables and Figures have been renamed to be consistent with CHAPTER 3.

Supplemental Information

Table S3-1 **DNA Oligonucleotides used in this study.**

Name	Used for	Sequence
ygaR	Set 1.850	g*c*gaagatcagtaaagatatagaaggtggtatccctggctattaAcaaggtcaggttttgattccattcattaaagatccagtaacaa*a*a
yqaC	Set 1.700	a*t*taaaaattatgatgggtccacgcgtgtcggcggtgaggcgtaActtaataaaggttgctctacctatcagcagctctacaatgaat*t*c
gabT	Set 1.600	t*c*accattgaagacgctcagatccgtcagggtctggagatcatcagccagtgttttgatgaggcgaagcagtaAcgccgctcctatgc*c*g
ygaU	Set 1.500	t*g*acgccaattcccattatccagcaggcgatggctggcaattaaTtacttcttccggaatacgcaacacttgccccggataaatttttat*c*c
ygaM	Set 1.400	g*t*aggtatttttatcggcgcactgttaagcatgcgcaaatcgtaAtgcaaaaatgataataaatacgcgtctttgaccccgaagcctg*t*c
luxS	Set 1.300	t*t*tgaactggctttttttcaattaattgtgaagatagtttactgaTtagatgtgcagttcctgcaacttctctttcggcagtgccagtt*c*t
mltB	Set 1.250	a*a*ttttacgaggaggattcagaaaaaagctgattagccagagggaagctcacgccccctcttgtaaatagTtactgtactcgcgcca*g*c
srlE	Set 1.200	a*c*tgtactgatcgcctggtttgtctccggttttatctatcaataAaggctgaaacatgaccgttatttatcagaccaccatcacccgt*a*t
norW	Set 1.150	a*t*cggatgaaagaggcatttggattgttgaaaacattgccgatgtaAgtgggctactgtgcctaaaatgtcggatgcgacgctggcgc*g*t
ascB	Set 1.100	a*t*cattctggtggtataaaaaagtgattgccagtaatggggaagatttagagtaAgtaacagtgccggatgcggcgtgaacgccttat*c*c
bioD	Set 2.850	t*c*gaagacgcgatctcgctcgcaatttaaccaaatacagaatggTtacaacaaggcaaggtttatgtactttccggttgccgcatttt*c*t
moaE	Set 2.700	c*g*taaacgtatgtactgagcggtgaaattgccggacgcagcggtgcctatccggctaacaaaaaaTtaccagcgttttgccgcctgc*t*g
ybhM	Set 2.600	g*c*gatgtgaagtttagttaagttctttagtatgtgcatttacggTtaatgaaaaaaacgcgtatgcctttgccagacaagcgttatag*c*t
ybhS	Set 2.500	t*t*tatcggcctgacgtggctgaaaaccaaacgtcggctggattaAggagaagagcatgtttcatcgcttatggacgttaatccgcaaa*g*a
ybiH	Set 2.400	c*a*tatcgacctgattttgcaaggattatcgcaaaggagtttgtaAtgatgaaaaaacctgtcgtgatcggattggcggtagtggtact*t*g
ybiR	Set 2.300	t*c*tgaattaatcttcaaaacttaaagcaaaaggcggactcataatccgccttttttatttgccagaccTtagttggccgggagtataa*c*t
yliD	Set 2.250	t*t*tcctgtgaggtgattacccttcaagcaatattcaaacgtaaTtatcttaattttcggatccagcgcatcgcgtaaaccatcgc*c*c
yliE	Set 2.200	g*a*ctgactgtaagtacgaacttattgattctggacatacgtaaaTtactcttttactaattttccacttttatcccaggcggagaatg*g*c
ybjK	Set 2.150	t*c*ggttcaaggttgatgggtttttttgttatctaaaacttatctaTtaccctgcaaccctctcaaccatcctcaaaatctcctcgcgcg*a*t
rimK	Set 2.100	c*g*caaaaagcgcaggcaaaaccatgatcagtaatgtgattgcgaTtaaccacccgttttcaggcaatattctgtcgtagcgtggcgtt*c*g

Table S3-1 (Continued).		
ygfJ	Set 3.850	c*c*ggacgactttattacagcgaaggaaaggtatactgaaatttaAaaaacgtagttaaacgattgcgttcaaatatttaatccttccg*g*c
recJ	Set 3.700	g*g*gattgtacccaatccacgctctttttatagagaagatgacgTtaaattggccagatattgtcgatgataatttgcaggctgcggt*t*g
argO	Set 3.600	c*t*ctggaggcaagcttagcgcctctgttttatttttccatcagatagcgcTtaactgaacaaggcttgtgcatgagcaataccgtctc*t*c
yggU	Set 3.500	a*a*tccgcaacaaatcccgccagaaatcgcggcgttaattaattaAgtatcctatgcaaaaagttgtcctcgcaaccggcaatgtcggt*a*a
mutY	Set 3.400	g*t*ggagcgtttgttacagcagttacgcactggcgcgccggtttaAcgcgtgagtcgataaagaggatgatttatgagcagaacgattt*t*t
glcC	Set 3.300	g*c*caccatttgattcgctcggcggtgccgctggagatgaacctgagttaActggtattaaatctgcttttcatacaatcggtaacgct*t*g
yghQ	Set 3.250	a*c*tgagtcagccgagaagaatttccccgcttattcgcaccttccTtaaatcaggtcatacgcttcgagatacttaacgccaaacacca*g*c
yghT	Set 3.200	t*g*gttgatgcagaaaaagcgattacggattttatgaccgcgcgtggttatcactaAtcaaaaatggaaatgcccgatcgccaggaccg*g*g
ygiZ	Set 3.150	t*t*ctctgtctatgagagccgttaaaacgactctcatagattttaTtaatagcaaaatataaaccgtccccaaaaaagccaccaaccac*a*a
yqiB	Set 3.100	a*g*ggttaacaggctttccaaatggtgtccttaggtttcacgacgTtaataaaccggaatcgccatcgctccatgtgctaaacagtatc*g*c
ygfJ_AGR	Set 3X.850	c*c*actatgtcagccatcgactgtataattaccgctgccggattatcatcaAGGatggggcaatggaaaatgatgttaccctgggaaca*g*g
ygfT_AGR	Set 3X.700	g*a*tgccttcgtatcaaacagagttaacatatcgcgcgccgcctgTCTtcctgcggccattgcagtgacaaccagatccgcgccatgaa*c*t
ubiH_AGR	Set 3X.600	g*t*gcagagtttgcgccgcattgcccaccagcacggtacgatgggtaatagaCCTggcggcgtgggttaacgccagcggataagcactg*c*g
argO_AGR	Set 3X.500	g*g*attcagccaggtcactgccaacatggtggcgataattttccaCCTgccttgcttcatgacttcggcgctggctaactcaatattac*t*g
yqgC_AGR	Set 3X.400	g*a*atcctgagaagcgccgagatgggtataacatcggcaggtatgcaaagcAGGgatgcagagtgcggggaacgaatcttcaccagaac*g*g
trmI_AGR	Set 3X.300	t*t*ttttacgcagacgacggctacggttctttgccattatttcacTCTctcgaacattaagtcccatactccgtgaccaagacgatgac*c*a
glcC_AGR	Set 3X.250	a*c*gatctgctcgacgttcgcgcattactggagggcgaatcggcaAGActggcggcaacgctgggaacgcaggctgattttgttgtgat*a*a
yghT_AGR	Set 3X.200	g*t*gaacatcttattaccgttgtcgaaaaatatggtgctgccgaaAGGgttcatttaggaaaacaggccggaaatgtcggtcgtgcagt*g*a
ygiZ_AGR	Set 3X.150	a*a*tacatataccaaaactcgaacatttcccgcataaagagtttCCTtaagataagaataataagtggcgtaagaagaaaaaatgctg*c*a
cpdA_AGR	Set 3X.100	c*t*tcgtgcttttgtgcaaacaggtgagtgtcggtaatttgtaaaatcctgacCCTggcctcaccagccagaggaagggttaacaggct*t*t
lacZ_KO1	Set lacZ jackpot +61	T*C*ACTGGCCGTCGTTTTACAACGTCGTGACTGGGAAAACCCTtGaGTTACCCAACTTAATCGCCTTGCAGCACATCCCCCTTTCGCCA*G*C

Table S3-1 (Continued).		
lacZ_KO2	Set lacZ jackpot +264	G*C*TGGAGTGCGATCTTCCTGAGGCCGATAC TGTCGTCGTCCCCTCAtAaTGGCAGATGCACG GTTACGATGCGCCCATCTACACCAAC*G*T
lacZ_KO3	Set lacZ jackpot +420	C*A*CATTTAATGTTGATGAAAGCTGGCTACA GGAAGGCCAGACGtaAATTATTTTTGATGGCG TTAACTCGGCGTTTCATCTGTGGTGC*A*A
lacZ_KO4	Set lacZ jackpot +602	T*G*ATGGTGCTGCGCTGGAGTGACGGCAGTT ATCTGGAAGATCAGtAgATGTGGCGGATGAGC GGCATTTTCCGTGACGTCTCGTTGCT*G*C
lacZ_KO5	Set lacZ jackpot +693	T*A*AACCGACTACACAAATCAGCGATTTCCA TGTTGCCACTCGCTaaAATGATGATTTCAGCC GCGCTGTACTGGAGGCTGAAGTTCAG*A*T
lacZ_KO6	Set lacZ jackpot +1258	T*A*CGGCCTGTATGTGGTGGATGAAGCCAAT ATTGAAACCCACtGaATGGTGCCAATGAATCG TCTGACCGATGATCCGCGCTGGCTAC*C*G
lacZ_KO7	Set lacZ jackpot +1420	G*G*GAATGAATCAGGCCACGGCGCTAATCA CGACGCGCTGTATtGaTGGATCAAATCTGTCG ATCCTTCCCGCCCGGTGCAGTATGAAG*G*C
lacZ_KO8	Set lacZ jackpot +1599	G*T*CCATCAAAAAATGGCTTTCGCTACCTGG AGAGACGCGCCCGtaGATCCTTTGCGAATACG CCCACGCGATGGGTAACAGTCTTGGC*G*G
lacZ_KO9	Set lacZ jackpot +1710	G*T*TTCGTCAGTATCCCCGTTTACAGGGCGG CTTCGTCTGGGACTaaGTGGATCAGTCGCTGA TTAAATATGATGAAAACGGCAACCCG*T*G
lacZ_KO10	Set lacZ jackpot +1890	A*G*CGCTGACGGAAGCAAAACACCAGCAGC AGTTTTTCCAGTTCtGaTTATCCGGGCAAACCA TCGAAGTGACCAGCGAATACCTGTTC*C*G
ygfJ_2*:2*_lead	Set 3.850_lead oligo	G*C*CGGAAGGATTAAATATTTGAACGCAAT CGTTTAACTACGTTTTTTAAATTTCAGTATAC CTTTCCTTCGCTGTAATAAAGTCGTCC*G*G
recJ_2*:2*_lead	Set 3.700_lead oligo	C*A*ACCGCAGCCTGCAAATTATCATCGACAA TATCTGGCCAATTTAACGTCATCTTCTCTATA AAAAAGAGCGTGGATTGGGTACAATC*C*C
argO_2*:2*_lead	Set 3.600_lead oligo	G*A*GAGACGGTATTGCTCATGCACAAGCCTT GTTCAGTTAAGCGCTATCTGATGGAAAAATA AAACAGAGGCGCTAAGCTTGCCTCCAG*A*G
yggU_2*:2*_lead	Set 3.500_lead oligo	T*T*ACCGACATTGCCGGTTGCGAGGACAACT TTTTGCATAGGATACTTAATTAATTAACGCCG CGATTTCTGGCGGGATTTGTTGCGGA*T*T
mutY_2*:2*_lead	Set 3.400_lead oligo	A*A*AAATCGTTCTGCTCATAAATCATCCTCT TTATCGACTCACGCGTTAAACCGGCGCGCCA GTGCGTAACTGCTGTAACAAACGCTCC*A*C
glcC_2*:2*_lead	Set 3.300_lead oligo	C*A*AGCGTTACCGATTGTATGAAAAGCAGA TTTAATACCAGTTAACTCAGGTTCATCTCCAG CGGCACCGCCGAGCGAATCAAATGGTG*G*C

Table S3-1 (Continued).		
yghQ_2*:2*_lead	Set 3.250_lead oligo	G*C*TGGTGTTTGGCGTTAAGTATCTCGAAGC GTATGACCTGATTTAAGGAAGGTGCGAATAA GCGGGGAAATTCTTCTCGGCTGACTCA*G*T
yghT_2*:2*_lead	Set 3.200_lead oligo	C*C*CGGTCCTGGCGATCGGGCATTTCCATTT TTGATTAGTGATAACCACGCGCGGTCATAAA ATCCGTAATCGCTTTTTCTGCATCAAC*C*A
ygiZ_2*:2*_lead	Set 3.150_lead oligo	T*T*GTGGTTGGTGGCTTTTTTGGGGACGGTT TATATTTTGCTATTAATAAAATCTATGAGAGT CGTTTTAACGGCTCTCATAGACAGAG*A*A
yqiB_2*:2*_lead	Set 3.100_lead oligo	G*C*GATACTGTTTAGCACATGGAGCGATGGC GATTCCGGTTTATTAACGTCGTGAAACCTAA GGACACCATTTGGAAAGCCTGTTAACC*C*T
exoX.KO*	exoX KO oligo	t*t*c*g*gcctggagcatgccatgttgcgcattatcgatacagaaacT GAtgcggtttgcagggagggatcgttgagattgcctctgttgatg
xseA.KO*	xseA KO oligo	g*a*a*t*ttgatctcgctcacatgttaccttctcaatcccctgcaatTGA tttaccgttagtcgcctgaatcaaacggttcgtctgctgcttg
recJ.KO*	recJ KO oligo	g*g*a*g*gcaattcagcgggcaagtctgccgtttcatcgacttcacgT CAcgacgaagttgtatctgttgtttcacgcgaattatttaccgct
xonA.KO*	xonA KO oligo	a*a*t*a*acggatttaacctaatgatgaatgacggtaagcaacaatcT GAacctttttgtttcacgattacgaaacctttggcacgcaccccg
Lexo.KO.MM*	Lambda exo KO oligo	t*g*a*a*acagaaagccgcagagcagaaggtggcagcatgacaccg taacattatcctgcagcgtaccgggatcgatgtgagagctgtcgaac
dnaG_Q576A	Oligo to make dnaG Q576A mutation	gcacgcatggtttaagcaacgaagaacgcctggagctctggacattaaac GCggaActggcgaaaaagtgatttaacggcttaagtgccg
dnaG_K580A	Oligo to make dnaG K580A mutation	cgcacgcatggtttaagcaacgaagaacgcctggagctctggacattaaa ccaggaActggcgGCaaagtgatttaacggcttaagtgcc
tolC.90.del	Oligo that deletes endogenous tolC	gaatttcagcgacgtttgactgccgtttgagcagtcatgtgttaaagcttcgg ccccgtctgaacgtaaggcaacgtaaagatacgggttat
galK_KO1.100	Oligo to delete 100 bp including a portion of galK	C*G*CGCAGTCAGCGATATCCATTTTCGCGAA TCCGGAGTGTAAGAAAACACACCGACTACAA CGACGGTTTCGTTCTGCCCTGCGCGAT*T*G
galK_KO1.1149	Oligo to delete 1149 bp including a portion of galK	C*G*CGCAGTCAGCGATATCCATTTTCGCGAA TCCGGAGTGTAAGAAACGAAACTCCCGCACT GGCACCCGATGGTCAGCCGTACCGACT*G*T
galK_KO1.7895	Oligo to delete 7895 bp including a portion of galK, galM, gpmA, aroG, ybgS, zitB, pnuC, and nadA	C*G*CGCAGTCAGCGATATCCATTTTCGCGAA TCCGGAGTGTAAGAACTTACCATCTCGTTTTA CAGGCTTAACGTTAAAACCGACATTA*G*C
ygaR_wt-f	Set 1.850_wt-f mascPCR	AAGGTGGTATCCCTGGCTATTAG
yqaC_wt-f	Set 1.700_wt-f mascPCR	CGGCGGTGAGGCGTAG
gabT_wt-f	Set 1.600_wt-f mascPCR	TTTTGATGAGGCGAAGCAGTAG

Table S3-1 (Continued).		
ygaU_wt-f	Set 1.500_wt-f mascPCR	GTTGCGTATTCCGGAAGAGTAG
ygaM_wt-f	Set 1.400_wt-f mascPCR	GTTAAGCATGCGCAAATCGTAG
luxS_wt-f	Set 1.300_wt-f mascPCR	GTTGCAGGAACTGCACATCTAG
mltB_wt-f	Set 1.250_wt-f mascPCR	GCTGGCGCGAGTACAGTAG
srlE_wt-f	Set 1.200_wt-f mascPCR	GGTTTGTCTCCGGTTTTATCTATCAATAG
norW_wt-f	Set 1.150_wt-f mascPCR	GATTGTTGAAAACATTGCCGATGTAG
ascB_wt-f	Set 1.100_wt-f mascPCR	CCAGTAATGGGGAAGATTTAGAGTAG
bioD_wt-f	Set 2.850_wt-f mascPCR	AGTACATAAACCTTGCCTTGTTGTAG
moaE_wt-f	Set 2.700_wt-f mascPCR	GCGGCAAAACGCTGGTAG
ybhM_wt-f	Set 2.600_wt-f mascPCR	AAGGCATACGCGTTTTTTTCATTAG
ybhS_wt-f	Set 2.500_wt-f mascPCR	CCAAACGTCGGCTGGATTAG
ybiH_wt-f	Set 2.400_wt-f mascPCR	AAGGATTATCGCAAAGGAGTTTGTAG
ybiR_wt-f	Set 2.300_wt-f mascPCR	TTAGTTATACTCCCGGCCAACTAG
yliD_wt-f	Set 2.250_wt-f mascPCR	CGCTGGATCCGAAAATTAAAGGATAG
yliE_wt-f	Set 2.200_wt-f mascPCR	TGGGATAAAAGTGGAAAATTAGTAAAAGAG TAG
ybjK_wt-f	Set 2.150_wt-f mascPCR	TTGAGAGGGTTGCAGGGTAG
rimK_wt-f	Set 2.100_wt-f mascPCR	GCCTGAAAACGGGTGGTTAG
ygfJ_wt-f	Set 3.850_wt-f mascPCR	AGCGAAGGAAAGGTATACTGAAATTTAG
recJ_wt-f	Set 3.700_wt-f mascPCR	TCATCGACAATATCTGGCCAATTTAG
argO_wt-f	Set 3.600_wt-f mascPCR	TGCACAAGCCTTGTTCAGTTAG
yggU_wt-f	Set 3.500_wt-f mascPCR	CAGAAATCGCGGCGTTAATTAATTAG
mutY_wt-f	Set 3.400_wt-f mascPCR	GGCGCGCCGGTTTAG

Table S3-1 (Continued).		
glcC_wt-f	Set 3.300_wt-f mascPCR	GCTGGAGATGAACCTGAGTTAG
yghQ_wt-f	Set 3.250_wt-f mascPCR	CTCGAAGCGTATGACCTGATTTAG
yghT_wt-f	Set 3.200_wt-f mascPCR	CGCGCGTGGTTATCACTAG
ygiZ_wt-f	Set 3.150_wt-f mascPCR	TGGGGACGGTTTATATTTTGCTATTAG
yqiB_wt-f	Set 3.100_wt-f mascPCR	CGATGGCGATTCCGGTTTATTAG
ygfJ_WT	Set 3X.850_wt-f mascPCR	GCTGCCGGATTATCATCAAGA
ygfT_WT	Set 3X.700_wt-f mascPCR	GCAATGGCCGCAGGAAGG
ubiH_WT	Set 3X.600_wt-f mascPCR	GCACGGTACGATGGGTAATAGAT
argO_WT	Set 3X.500_wt-f mascPCR	GAAGTCATGAAGCAAGGCAGA
yqgC_WT	Set 3X.400_wt-f mascPCR	CGGCAGGTATGCAAAGCAGA
trmI_WT	Set 3X.300_wt-f mascPCR	AGTATGGGACTTAATGTTCGAGAGG
glcC_WT	Set 3X.250_wt-f mascPCR	AGGGCGAATCGGCAAGG
yghT_WT	Set 3X.200_wt-f mascPCR	GAAAAATATGGTGCTGCCGAAAGA
ygiZ_WT	Set 3X.150_wt-f mascPCR	CTTCTTACGCCACTTATTATTCTTATCTTAAGA
cpdA_WT	Set 3X.100_wt-f mascPCR	TGGCTGGTGAGGCCAGA
exoX.KO*-wt-f	exoX wt-f mascPCR primer	GCGCATTATCGATACAGAAACCT
xseA.KO*-wt-f	xseA wt-f mascPCR primer	CTTCTCAATCCCCTGCAATTTTTACC
recJ.KO*-wt-f	recJ wt-f mascPCR primer	CAACAGATACAACTTCGTCGCC
xonA.KO*-wt-f	xonA wt-f mascPCR primer	GAATGACGGTAAGCAACAATCTACC
Lexo_WT-f	Lambda exo KO wt-f mascPCR primer	GGCAGCATGACACCGGA
dnaG_Q576A_wt-f	dnaG_Q576A wt-f mascPCR primer	TGGAGCTCTGGACATTAAACCA
dnaG_K580A_wt-f	dnaG_K580A wt-f mascPCR primer	CATTAAACCAGGAACTGGCGAA

Table S3-1 (Continued).		
ygaR_mut-f	Set 1.850_mut-f mascPCR	AAGGTGGTATCCCTGGCTATTAA
yqaC_mut-f	Set 1.700_mut-f mascPCR	CGGCGGTGAGGCGTAA
gabT_mut-f	Set 1.600_mut-f mascPCR	TTTTGATGAGGCGAAGCAGTAA
ygaU_mut-f	Set 1.500_mut-f mascPCR	GTTGCGTATTCCGGAAGAGTAA
ygaM_mut-f	Set 1.400_mut-f mascPCR	GTTAAGCATGCGCAAATCGTAA
luxS_mut-f	Set 1.300_mut-f mascPCR	GTTGCAGGAACTGCACATCTAA
mltB_mut-f	Set 1.250_mut-f mascPCR	GCTGGCGCGAGTACAGTAA
srlE_mut-f	Set 1.200_mut-f mascPCR	GGTTTGTCTCCGGTTTTATCTATCAATAA
norW_mut-f	Set 1.150_mut-f mascPCR	GATTGTTGAAAACATTGCCGATGTAA
ascB_mut-f	Set 1.100_mut-f mascPCR	CCAGTAATGGGGAAGATTTAGAGTAA
bioD_mut-f	Set 2.850_mut-f mascPCR	AGTACATAAACCTTGCCTTGTTGTAA
moaE_mut-f	Set 2.700_mut-f mascPCR	GCGGCAAAACGCTGGTAA
ybhM_mut-f	Set 2.600_mut-f mascPCR	AAGGCATACGCGTTTTTTTCATTAA
ybhS_mut-f	Set 2.500_mut-f mascPCR	CCAAACGTCGGCTGGATTAA
ybiH_mut-f	Set 2.400_mut-f mascPCR	AAGGATTATCGCAAAGGAGTTTGTAA
ybiR_mut-f	Set 2.300_mut-f mascPCR	TTAGTTATACTCCCGGCCAACTAA
yliD_mut-f	Set 2.250_mut-f mascPCR	CGCTGGATCCGAAAATTAAAGGATAA
yliE_mut-f	Set 2.200_mut-f mascPCR	TGGGATAAAAGTGGAAAATTAGTAAAAGAG TAA
ybjK_mut-f	Set 2.150_mut-f mascPCR	TTGAGAGGGTTGCAGGGTAA
rimK_mut-f	Set 2.100_mut-f mascPCR	GCCTGAAAACGGGTGGTTAA
ygfJ_mut-f	Set 3.850_mut-f mascPCR	AGCGAAGGAAAGGTATACTGAAATTTAA
recJ_mut-f	Set 3.700_mut-f mascPCR	TCATCGACAATATCTGGCCAATTTAA

Table S3-1 (Continued).		
argO_mut-f	Set 3.600_mut-f mascPCR	TGCACAAGCCTTGTTCAGTTAA
yggU_mut-f	Set 3.500_mut-f mascPCR	CAGAAATCGCGGCGTTAATTAATTAA
mutY_mut-f	Set 3.400_mut-f mascPCR	GGCGCGCCGGTTTAA
glcC_mut-f	Set 3.300_mut-f mascPCR	GCTGGAGATGAACCTGAGTTAA
yghQ_mut-f	Set 3.250_mut-f mascPCR	CTCGAAGCGTATGACCTGATTTAA
yghT_mut-f	Set 3.200_mut-f mascPCR	CGCGCGTGGTTATCACTAA
ygiZ_mut-f	Set 3.150_mut-f mascPCR	TGGGGACGGTTTATATTTTGCTATTAA
yqiB_mut-f	Set 3.100_mut-f mascPCR	CGATGGCGATTCCGGTTTATTAA
ygfJ_MUT	Set 3X.850_mut-f mascPCR	GCTGCCGGATTATCATCAAGG
ygfT_MUT	Set 3X.700_mut-f mascPCR	GCAATGGCCGCAGGAAGA
ubiH_MUT	Set 3X.600_mut-f mascPCR	GCACGGTACGATGGGTAATAGAC
argO_MUT	Set 3X.500_mut-f mascPCR	GAAGTCATGAAGCAAGGCAGG
yqgC_MUT	Set 3X.400_mut-f mascPCR	GGCAGGTATGCAAAGCAGG
trmI_MUT	Set 3X.300_mut-f mascPCR	GAGTATGGGACTTAATGTTCGAGAGA
glcC_MUT	Set 3X.250_mut-f mascPCR	GAGGGCGAATCGGCAAGA
yghT_MUT	Set 3X.200_mut-f mascPCR	AAAATATGGTGCTGCCGAAAGG
ygiZ_MUT	Set 3X.150_mut-f mascPCR	CTTCTTACGCCACTTATTATTCTTATCTTAAGG
cpdA_MUT	Set 3X.100_mut-f mascPCR	GGCTGGTGAGGCCAGG
exoX.KO*-mut-f	exoX mut-f mascPCR primer	GCGCATTATCGATACAGAAACTGA
xseA.KO*-mut-f	xseA mut-f mascPCR primer	CTTCTCAATCCCCTGCAATTGA
recJ.KO*-mut-f	recJ mut-f mascPCR primer	CAACAGATACAACTTCGTCGTGA
xonA.KO*-mut-f	xonA mut-f mascPCR primer	GAATGACGGTAAGCAACAATCTGA

Table S3-1 (Continued).		
Lexo_MUT-f	Lambda exo KO mut-f mascPCR primer	TGGCAGCATGACACCGTAA
dnaG_Q576A_mut-f	dnaG_Q576A mut-f mascPCR primer	GGAGCTCTGGACATTAAACGC
dnaG_K580A_mut-f	dnaG_K580A mut-f mascPCR primer	ACCAGGAACTGGCGGC
ygaR_rev	Set 1.850_rev mascPCR	TAGGTAGAGCAACCTTTATTAAGCTACG
yqaC_rev	Set 1.700_rev mascPCR	TAAAAATATCTACATTTCTGAAAAATGCGCA
gabT_rev	Set 1.600_rev mascPCR	GCGGCGATGTTGGCTT
ygaU_rev	Set 1.500_rev mascPCR	AGGGTATCGGGTGGCG
ygaM_rev	Set 1.400_rev mascPCR	CGCAACGCTTCTGCCG
luxS_rev	Set 1.300_rev mascPCR	ATGCCCAGGCGATGTACA
mltB_rev	Set 1.250_rev mascPCR	AGACTCGGCAGTTGTTACGG
srlE_rev	Set 1.200_rev mascPCR	GGATGGAGTGCACCTTTCAAC
norW_rev	Set 1.150_rev mascPCR	GTGTTGCATTTGGACACCATTG
ascB_rev	Set 1.100_rev mascPCR	CGCTTATCGGGCCTTCATG
bioD_rev	Set 2.850_rev mascPCR	CGGGAAGAACTCTTTCATTTCGC
moaE_rev	Set 2.700_rev mascPCR	CGTCAATCCGACAAAGACAATCA
ybhM_rev	Set 2.600_rev mascPCR	TTACTGGCAGGGATTATCTTTACCG
ybhS_rev	Set 2.500_rev mascPCR	CTGTTGTTAGGTTTCGGTTTTCCT
ybiH_rev	Set 2.400_rev mascPCR	GTCATAGGCGGCTTGCG
ybiR_rev	Set 2.300_rev mascPCR	ATGAGCCGGTAAAAGCGAC
yliD_rev	Set 2.250_rev mascPCR	AATAAAATTATCAGCCTTATCTTTATCTTTTCGTATAAA
yliE_rev	Set 2.200_rev mascPCR	CAGCAATATTTGCCACCGCA
ybjK_rev	Set 2.150_rev mascPCR	AACTTTTCCGCAGGGCATC

Table S3-1 (Continued).		
rimK_rev	Set 2.100_rev mascPCR	TACAACCTCTTTCGATAAAAAGACCG
ygfJ_rev	Set 3.850_rev mascPCR	GATGAACTGTTGCATCGGCG
recJ_rev	Set 3.700_rev mascPCR	CTGTACGCAGCCAGCC
argO_rev	Set 3.600_rev mascPCR	AATCGCTGCCTTACGCG
yggU_rev	Set 3.500_rev mascPCR	TAACCAAAGCCACCAGTGC
mutY_rev	Set 3.400_rev mascPCR	CGCGAGATATTTTTTCATCATTCCG
glcC_rev	Set 3.300_rev mascPCR	GGGCAAAATTGCTGTGGC
yghQ_rev	Set 3.250_rev mascPCR	ACCAACTGGCGATGTTATTCAC
yghT_rev	Set 3.200_rev mascPCR	GACGATGGTGGTGGACGG
ygiZ_rev	Set 3.150_rev mascPCR	ATCGCCAAATTGCATGGCA
yqiB_rev	Set 3.100_rev mascPCR	AAAATCCTGACTCTGGCCTCA
ygfJ_rev	Set 3X.850_rev mascPCR	TCTGTTTGCACTGCGGGTAC
ygfT_rev	Set 3X.700_rev mascPCR	TGGTTGGGCAATCTAATAGATTCTCC
ubiH_rev	Set 3X.600_rev mascPCR	atgAGCGTAATCATCGTCGGTG
argO_rev	Set 3X.500_rev mascPCR	CCGTCTCTCGCCAGCTG
yqgC_rev	Set 3X.400_rev mascPCR	AGCACACGACGTTTCTTTCG
trmI_rev	Set 3X.300_rev mascPCR	ATCTGTTCTTCCGATGTACCTTCC
glcC_rev	Set 3X.250_rev mascPCR	CTTCCAGCTCGATATCGTGGAG
yghT_rev	Set 3X.200_rev mascPCR	CACCACCAAAGGTTAACTGTGG
ygiZ_rev	Set 3X.150_rev mascPCR	CACAAACCAGACAAATACCGAGC
cpdA_rev	Set 3X.100_rev mascPCR	CGATGGTATCCAGCGTAAAGTTG
exoX.KO*-r	exoX rev mascPCR primer	GACCATGGCTTCGGTGATG

Table S3-1 (Continued).		
xseA.KO*-r	xseA rev mascPCR primer	GGTACGCTTAAGTTGATTTTCCAGC
recJ.KO*-r	recJ rev mascPCR primer	GGCCTGATCGACCACTTCC
xonA.KO*-r	xonA rev mascPCR primer	GAAATGTCTCCTGCCAAATCCAC
Lexo-r	Lambda exo KO rev mascPCR primer	CAAGGCCGTTGCCGTC
dnaG_seq-r	dnaG rev mascPCR primer for both Q576A and K580A	GCTCCATAAGACGGTATCCACA
Rx-P19	forward screening primer for wt tolC deletion	GTTTCTCGTGCAATAATTTCTACATC
Rx-P20	reverse screening primer for wt tolC deletion	CGTATGGATTTTGTCCGTTTCA
lacZ_jackpot_seq-f	forward sequencing primer for lacZ jackpot alleles	GAATTGTGAGCGGATAACAATTTC
lacZ_jackpot_seq-r	reverse sequencing primer for lacZ jackpot alleles	CCAGCGGCTTACCATCC
cat_mut*	cat inactivation oligo	G*C*ATCGTAAAGAACATTTTGAGGCATTTCAGTCAGTTGCTTAATGTACCTATAACCAGACCGTTCAGCTGGATATTACGGCCTTTTTA*A*A
cat_restore*	cat reactivation oligo	G*C*ATCGTAAAGAACATTTTGAGGCATTTCAGTCAGTTGCTCAATGTACCTATAACCAGACCGTTCAGCTGGATATTACGGCCTTTTTA*A*A
tolC-r_null_mut*	tolC inactivation oligo	A*G*CAAGCACGCCTTAGTAACCCGGAATTGCGTAAGTCTGCCGCTAAATCGTGATGCTGCCTTTGAAAAAATTAATGAAGCGCGCAGTCCA
tolC-r_null_revert*	tolC reactivation oligo	C*A*GCAAGCACGCCTTAGTAACCCGGAATTGCGTAAGTCTGCCGCCGATCGTGATGCTGCCTTTGAAAAAATTAATGAAGCGCGCAGTCCA
tolC_null_revert*	tolC reactivation oligo (leading targeting)	T*G*GACTGCGCGCTTCATTAATTTTTTCAAAGGCAGCATCACGATCGGCGGCAGACTTACGCAATTCCGGGTTACTAAGGCGTGCTTGCTG
bla_mut*	bla inactivation oligo	G*C*C*A*CATAGCAGAACTTTAAAAGTGCTCATCATTGGAAAACGTTATTAGGGGCGAAAACTCTCAAGGATCTTACCGCTGTTGAGATCCAG
bla_restore*	bla reactivation oligo	G*C*C*A*CATAGCAGAACTTTAAAAGTGCTCATCATTGGAAAACGTTCTTCGGGGCGAAAACTCTCAAGGATCTTACCGCTGTTGAGATCCAG
313000.T.lacZ.coMAGE-f	Cassette primer for T.co-lacZ (lacZ coselection)	TGCTTCTCATGAACGATAACACAACTTGTTCATGAATTAACCATTCCGGATTGAGGCACATTAACGCC
313001.T.lacZ.coMAGE-r	Cassette primer for T.co-lacZ (lacZ coselection)	ACGGAAACCAGCCAGTTCCTTTCGATGCCTGAATTTGATCCCATAGTTTATCTAGGGCGGCGGATT
312869.seq-f	Screening primer for tolC (lacZ coselection)	GAACTTGCACTACCCATCG

Table S3-1 (Continued).		
313126.seq-r	Screening primer for tolC (lacZ coselection)	AGTGACGGGTTAATTATCTGAAAG
1255700.S.12.13b-f	Cassette primer for S.12.13b	TTTCATCTTGCCAGCATATTGGAGCGTGATCAATTTTGATCAGCTGTGAACAGCCAGGACAGAAATGC
1255701.S.12.13b -r	Cassette primer for S.12.13b	CATTAGCAGTGATATAACGTAAGTTTTTGTATCACTACACATCAGCCCCCTGCAGAAATAAAAAGGCCTGC
1255550.Seq-f	Screening primer for S.12.13b	CATTTTTGCATTACTAATAAGAAAAAGCAAA
1255850.Seq-r	Screening primer for S.12.13b	GTCCTAATCATTCTTGTAACATCCTAC
1710450.Z.16.17b-f	Cassette primer for Z.16.17b	TCAGGTTAAAATCATTTAAATTTACACACGCAACAAATATTGACCTACAAGGTGTTGACAATTAATCATCGGC
1710451.Z.16.17b-r	Cassette primer for Z.16.17b	TTTTTACTAGTGAGATAGTCCAGTTTCTGAAAAATAGCCAGTGTAATGTTAGCTTGCAAATTAAAGCCTTCG
1710300.Seq-f	Screening primer for Z.16.17b	TCAGGTAATCCGTTTGCGG
1710600.Seq-r	Screening primer for Z.16.17b	AACGGCAGATTTTTTCACTGC
LacZ::KanR.full-f	Cassette primer for lacZ::kanR	TGACCATGATTACGGATTCACTGGCCGTCGTTTTACAACGTCGTGCCTGTGACGGAAGATCACTTCG
LacZ::KanR.full-r	Cassette primer for lacZ::kanR	GTGCTGCAAGGCGATTAAGTTGGGTAACGCCAGGGTTTTCCCAGTAACCAGCAATAGACATAAGCGG

An asterisk (*) indicates use of a phosphorothioate bond to protect against exonuclease activity (1).

Table S3-2: Estimation of Okazaki fragment length in EcNR2.dnaG.K580A and EcNR2.dnaG.Q576A

[Primase] (nM)	WT dnaG Okazaki Frag (kb)	K580A Okazaki Frag (kb)	Q576A Okazaki Frag (kb)
80	2.5	5	23
160	1.5	2.5	18
320	1	1	8
640	0.8	nd	3
Average Fold effect compared to WT		**1.6**	**8.2**

Average Okazaki Fragment length was estimated based on *in vitro* results (gel images) from Tougu et al. (2) for the same DnaG primase variants, tabulated above. We compared the fold difference in OF sizes for the specified primase concentrations and determined the average fold difference (variant OF length/wt OF length). We estimate *in vivo* OF lengths of ~2.3-3.1 kb and ~12-16 kb for the K580A and Q576A mutants, respectively, based on the reported ~1.5-2 kb OF lengths in wt cells grown in rich media (3-5). However, these approximations may be imperfect since Tougu et al. (2) performed this analysis *in vitro* and did not use the same EcNR2.dnaG.K580A and EcNR2.dnaG.Q576A strains. Other conditions and/or host factors not

accounted for *in vitro* may affect priming efficiency, thereby rendering these calculations inaccurate.

Supplemental Figures

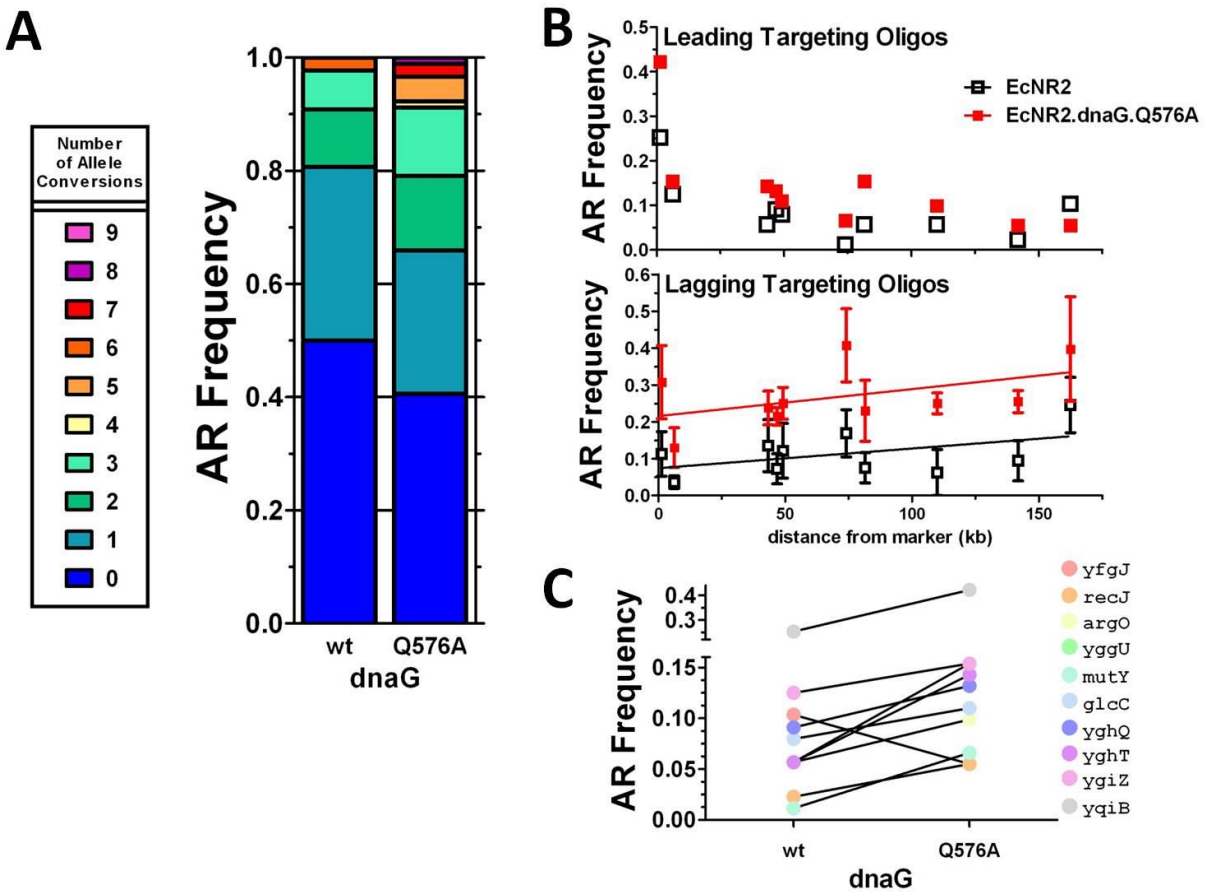


Figure S3-1. Effect of dnaG variants and co-selection on leading-targeting CoS-MAGE. **(A)** EcNR2.dnaG.Q576A (n=91) outperformed EcNR2 (wt, n=88) in leading-targeting Set 3 CoS-MAGE, with a reduction in zero conversion events as well as a broadening of the distribution of total allele conversions per clone. **(B)** For leading-targeting Set 3 oligos, AR frequency decays rapidly with increasing distance from the selectable marker (top panel). In contrast, co-selection using the corresponding set of lagging targeting oligos (see also Figure 3C, right panel) provides robust co-selection spanning at least 0.162 Mb (bottom panel). For the lagging-targeting oligos (bottom panel), linear regression analyses (solid trendline) show that co-selection does not decrease with distance for either strain over this 0.162 Mb genomic region. **(C)** Individual CoS-MAGE AR frequency is plotted for each leading-targeting Set 3 oligo in EcNR2 (wt) and EcNR2.dnaG.Q576A (Q576A). AR frequency is improved for 9/10 alleles in EcNR2.dnaG.Q576A. Note that the most proximal allele to the selectable marker (*yqiB*) is separated from the other alleles with a broken axis, since its AR frequency was much higher than that of the others.

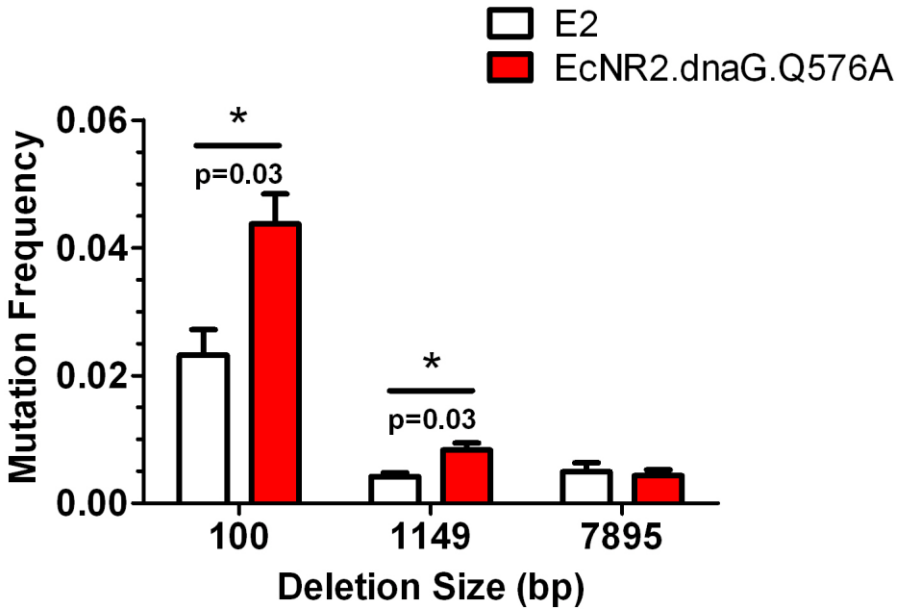


Figure S3-2. Effect of dnaG attenuation on deletion frequency. DnaG primase disruption enhances gene-sized deletion frequency. Oligos that deleted 100 bp, 1,149 bp, or 7,895 bp of the genome, including a portion of *galK*, were recombined into EcNR2 and EcNR2.dnaG.Q576A. The recombined populations were screened for the GalK- phenotype (white colonies) on MacConkey agar plates supplemented with galactose as a carbon source. EcNR2.dnaG.Q576A significantly outperformed EcNR2 for the 100 bp and 1,149 bp deletions, but there was no difference detected between the two strains for the 7,895 bp deletion.

Supplemental References

1. Wang, H.H., Isaacs, F.J., Carr, P.A., Sun, Z.Z., Xu, G., Forest, C.R. and Church, G.M. (2009) Programming cells by multiplex genome engineering and accelerated evolution. *Nature*, **460**, 894-898.
2. Tougu, K. and Marians, K.J. (1996) The Interaction between Helicase and Primase Sets the Replication Fork Clock. *Journal of Biological Chemistry*, **271**, 21398-21405.
3. Corn, J.E. and Berger, J.M. (2006) Regulation of bacterial priming and daughter strand synthesis through helicase-primase interactions. *Nucleic Acids Res.*, **34**, 4082-4088.
4. Lia, G., Michel, B. and Allemand, J.-F. (2012) Polymerase Exchange During Okazaki Fragment Synthesis Observed in Living Cells. *Science*, **335**, 328-331.
5. Okazaki, R., Okazaki, T., Sakabe, K., Sugimoto, K. and Sugino, A. (1968) Mechanism of DNA chain growth. I. Possible discontinuity and unusual secondary structure of newly synthesized chains. *Proceedings of the National Academy of Sciences*, **59**, 598-605.

APPENDIX C

Supplemental Material for Genome-wide Codon Replacement Using Synthetic Oligonucleotides and Engineered Conjugation

This supplemental material is reproduced with permission from its initial publication:

Isaacs FJ[*], Carr PA[*], Wang HH[*], **Lajoie MJ**, Sterling B, Kraal L, Tolonen AC, Gianoulis TA, Goodman DB, Reppas NB, Emig CJ, Bang D, Hwang SJ, Jewett MC, Jacobson JM, Church GM (2011) *Genome-wide Codon Replacement Using Synthetic Oligonucleotides and Engineered Conjugation*. **Science**: 333, 348-353.

Tables and Figures have been renamed to be consistent with CHAPTER 4.

Supporting Online Material:

Table of Contents:

I. Materials and Methods
II. Figure S4-1. Circular *E. coli* genome containing all TAG codon positions
III. Figure S4-2. Multiplexing in MAGE leads to higher aggregate allele replacement efficiencies
IV. Figure S4-3. Multiplex allele-specific colony quantitative PCR (MASC-qPCR) for the detection of advanced clones
V. Figure S4-4. Multiplex allele-specific colony PCR (MASC-PCR) for the enumeration of site conversions in each clone
VI. Figure S4-5. Off-target mutation in modified strains
VII. Figure S4-6. Growth rates of top modified (10-site) clones following MAGE cycling and selection
VIII. Figure S4-7. Damaged oligo**s** can be fixed by host repair mechanisms
IX. Figure S4-8. Bioinformatics sequencing analysis process
X. Figure S4-9. Functional annotation of SNPs
XI. Table S4-1. List and genomic coordinates of genes containing TAG codons
XII. Table S4-2. TAG codons in overlapping genes
XIII. Table S4-3. List of all TAG mutator oligos
XIV. Table S4-4. Frequency of TAG codon conversions at each locus
XV. Table S4-5. Frequency of TAG codon conversions overall
XVI. Table S4-6. Properties of MAGE-cycled strains
XVII. Table S4-7. Five-stage hierarchical conjugation process
XVIII. Table S4-8. List of all secondary mutations
XIX. Table S4-9. Tot**a**l number and type of SNPs observed across three genomes
XX. Table S4-10. Calculation of secondary mutation rates
XXI. Table S4-11. Estimated total number of cell divisions
XXII. Table S4-12. List of primers used in MASC-PCR reactions
XXIII. Table S4-13. List of primers used MASC-qPCR reactions

Materials & Methods

Strains and Culture Conditions

The λ-prophage was obtained from strain DY330 (*1*), modified to include the *bla* gene and introduced into wild-type MG1655 *E.coli* by P1 Transduction at the *bioA*/*bioB* gene locus and selected on ampicillin to yield the strain EcNR1 (λ-Red^{+}). Replacement of *mutS* with the chloramphenicol resistance gene (*chloramphenicol acetyl transferase, cmR*) in EcNR1 produced EcNR2 (mutS^{-}, λ-Red^{+}). EcNR2 was grown in low salt LB-min medium (10 g tryptone, 5 g yeast extract, 5 g NaCl in 1 L dH_2O) for optimal electroporation efficiency and compatibility with zeocin selection. EcNR2 was used as the ancestral strain for all recoded strains reported in this manuscript.

Oligonucleotides

All oligonucleotides were obtained from Integrated DNA Technologies. Oligonucleotides (Table S4-3) used in the MAGE process were designed according to the following specifications: 1) 90 nucleotides in length, 2) contain a single mutation to effect the TAG to TAA codon conversion, 3) two phosphorothioate linkages at both the 5' and 3' ends to attenuate exonuclease activity and to increase half life, 4) minimize secondary structure (ΔG threshold values, self-folding energy), 5) target lagging strand at the replication fork. No additional purification was used following oligonucleotide synthesis. Primers were purchased from IDT for the multiplex PCR assays and loci sequencing reactions (see description below and Tables S4-12 and S4-13).

MAGE-generated Codon Conversions

A single clone of the EcNR2 strain was grown in liquid cell culture, which was used to inoculate 32 separate cultures for parallel modification of all TAG codons. Modification of these codons was achieved through continuous MAGE (*2*) cycling. Each culture was grown at 30°C to mid-logarithmic growth (*i.e.,* OD_{600} of ~0.7) in a rotor drum at 200 RPM. To induce expression of the λ Red recombination proteins (Exo, Beta and Gam), cell cultures were shifted to a 42°C water bath with vigorous shaking for 15 min and then immediately chilled on ice. In a 4°C environment, 1 mL of cell culture was centrifuged at 16,000x g for 30 seconds. Supernatant media was removed and cells were re-suspended in 1 mL dH_2O (Gibco cat# 15230). This wash process was repeated. Supernatant water was removed the pellet was resuspended in the appropriate pool of 10 oligos (1 uM per MAGE oligo in 50 uL dH_2O). The resuspended oligos/cell mixture was transferred to a pre-chilled 96-well, 2 mm gap electroporation plate (BTX, USA) and electroporated with a BTX electroporation system using the following parameters: 2.5 kV, 200 Ω, and 25 μF. The electroporated cells were immediately transferred to 3 mL of LB-min media for recovery. Recovery cultures were grown at 30 °C in a rotator drum for 2-2.5 hours. Once cells reached mid-logarithmic growth they proceeded to the next MAGE cycle. This approach introduces genomic modifications while allowing cells to evolve and adapt to those changes. Moreover, this approach is designed to explore extensive genotype and phenotype landscapes by creating combinatorial genomic variants that leverage the size of the cell population. After 18 MAGE cycles, cells from each population were isolated on LB-min agar plates. Forty-seven clones from each of the 32 cycled populations were selected and subjected to genotype and phenotype analyses. From each population the clone with the greatest number of modifications (an average of 8 modifications per clone) and minimal aberrant

phenotypes (*i.e.*, auxotrophy, decreased fitness) was selected. Further MAGE cycles were employed (typically 6 cycles, but in some cases up to 15) to yield strains with complete sets of 10 targeted modifications.

Genotype Analyses

TAG-to-TAA codon conversions were analyzed using three main methods: 1) Multiplex allele-specific colony PCR (MASC-PCR), 2) Multiplex allele-specific colony quantitative PCR (MASC-qPCR) and 3) Sanger DNA sequencing.

Multiplex Allele Specific Colony PCR (MASC-PCR)

Based on previously described allele-specific PCR techniques, we developed the MASC-PCR method to test for TAG-to-TAA codon conversion in our recoded strains (the ancestral EcNR2 strain was the negative control). Three primers were designed for each locus: 1) a forward primer for the TAG sequence, 2) a forward primer for the TAA sequence and 3) a reverse primer compatible with both forward primers (Table S4-12). Primers were designed for a target T_m of 62° C. The two forward primers were identical except that the most 3' nucleotide hybridized to produce either a GC base pair for the wildtype (TAG) codon or an AT base pair for the mutant (TAA) codon. Thus, every clone from each of the 32 populations was interrogated via two MASC-PCR reactions, in which each reaction assayed 10 different loci (with one set assaying four loci). One reaction assayed the wild type (TAG) sequence and a second reaction assayed the mutant (TAA) sequence, yielding two binary reactions that revealed the sequences of the targeted codons (Figure S4-4). A clone containing the mutant allele generated PCR products only using the mutant allele primers and not the WT primers and vice versa for a clone with the wild-type allele. To minimize nonspecific amplification of MASC-PCR primers, a gradient PCR

was performed to experimentally determine the optimal annealing temperature for each MASC-PCR primer pool (typically between 64 - 67° C). Multiple loci were queried in a single PCR reaction using the multiplex PCR mast2er mix kit from Qiagen. Each MASC-PCR primer set produced amplicon lengths of 100, 150, 200, 250, 300, 400, 500, 600, 700, or 850 bps, corresponding to up to 10 different genomic loci. We found that using a 1:100 dilution of saturated clonal culture in water as template generated the best MASC-PCR specificity. Typical 20 uL MASC-PCR reactions included 10 uL 2x Qiagen multiplex PCR master mix, 0.2 uM of each primer, and 1 uL of template. MASC-PCR cycles were conducted as follows: polymerase heat activation and cell lysis for 15 min at 95° C, denaturing for 30 sec at 94° C, annealing for 30 sec at experimentally determined optimal temperature (64-67° C), extension for 80 sec at 72°C, repeated cycling 26 times, and final extension for 5 min at 72°C. Gel electrophoresis on a 1.5% agarose gel (0.5x TBE) produced the best separation for a 10-plex MASC-PCR reaction. (See Figure S4-4 for representative gel picture of MASC-PCR reaction.)

Mulitplex allele-specific quantitative colony PCR (MASC-qPCR)

In complement to MASC-PCR analyses, we also developed a highly multiplexed quantitative PCR screen to rapidly identify highly modified clones (Figure S4-3). Typical multiplexed qPCR reactions employ multiple fluorescence and distinct detection events to assess multiple PCR reactions in one sample, and are generally limited by the available optics and fluorescence to 4 channels. Instead, we needed a robust, economical test that employed many different non-optimized primers, did not require more expensive fluorescently labeled oligos, and would work for 10-plex reactions. We accomplished these goals with SYBR Green I detection, which gauges the total amount of DNA produced in the reaction. Two qPCR reactions

were compared for each clone evaluated, one with 10 pairs of primers matched to the unmodified TAG genes, and the other with 10 primer pairs matched to the intended TAA modifications. The TAG reactions were expected to proceed most efficiently with a wild-type template, and the TAA reactions most efficiently with a fully modified template. Intermediate values between these extremes also provided an effective, though nonlinear gauge of the extent of modification for each clone (Figure S4-3 A-C).

Each colony was used as template for a pair of qPCR reactions comparing the amplification efficiency when matched to primers terminating in wild-type or targeted mutant sequence. The experimental measurement for a given clone is then compared to the equivalent values measured for the unmodified starting (negative control) strain. This reference value is subtracted from each ΔC_t to yield a $\Delta\Delta C_t$, with unmodified clones scoring close to zero (as with the negative control colonies). The largest $\Delta\Delta C_t$ values were expected to indicate the most modified clones, which we confirmed by genotyping clones with varying $\Delta\Delta C_t$ values (Figure S4-3C) Large numbers of clones could be quickly assessed using this approach (up to 190 per 384-well plate, plus 2 negative controls). A typical assessment of MAGE-cycled clones comprised of 4 groups per plate, i.e. for each culture targeting 10 modifications, 2-4 control colonies and 44-46 queried colonies. After identification of the most promising clones, site-specific qPCR genotyping (Figure S4-3D) was used to identify which specific sites had been modified, selecting the best clones for further modification.

Individual bacterial colonies were picked into 0.5 mL sterile distilled deionized water, with 5 µL of this suspension used as template in 20 µL qPCR reactions containing 1x NovaTaq buffer, 0.5 U NovaTaq Hotstart DNA Polymerase (EMD Biosciences), 250 µM each dNTP, 0.5x SYBR Green I (Invitrogen), and 5% DMSO. Primer concentrations were 50 nM for each primer

(i.*e*. 500 nM total for 10 forward primers and 500 nM total for 10 reverse primers). A typical qPCR program included a 10 minute hot start at 95° C, followed by 40 cycles (95° C for 30 seconds, 60° C for 30 seconds, 72° C for 30 seconds) finishing with a melt curve analysis. All reactions were performed in a 7900 HT system (Applied Biosystems, Inc.). PCR primers for all sites were designed to have a melting temperature estimated at 62° C. Reverse primers were chosen to yield amplicons in the size range of 200-225 bp. No optimization was needed for qPCR primer sequences or for multiplex/singleplex reaction conditions.

Sanger Sequencing of 314 TAG to TAA loci

DNA sequencing was employed to confirm the results of the above PCR assays and to determine genotypes for 16 sites that gave ambiguous results by MASC-PCR. Amplicons 200-300 bp in length surrounding each of the 314 TAG sites were sequenced from the top-scoring clones by colony PCR as above. Sanger sequencing to confirm allelic replacements was performed by Agencourt Bioscience Corporation and the Biopolymer Facility in the Department of Genetics at Harvard Medical School. Mutations were identified by sequence alignment to the reference MG1655 genome.

Phenotype Analyses

To ensure that the codon replacements did not introduce any significant aberrant phenotypes, we conducted a number of experiments that assessed the fitness of the recoded strains. These experiments included measurements of: 1) strain growth rates, 2) auxotrophic rates and 3) frequency of recombination. Growth rate measurements were obtained by growing replicates of the recoded strains in LB-min media in 96-well plates at 30°C and obtaining OD_{600}

measurements using Molecular Devices plate readers (M5 and SpectraMax Plus). Auxotrophic rates were obtained by spotting all clonal isolates (1504) from the MAGE-cycled experiments on M9 minimal media plates (200 mL 5x M9 medium, 1 mL 1 M $MgSO_4$, 5 mL 40% glucose, 100 µL 0.5% vitamin B1 (thiamine), 1 mL D-biotin (0.25 mg/mL), up to 1 L water + 15g Agar). The recombination frequency of each isolate was obtained by performing the allelic replacement protocol using a lacZ 90-mer oligo that produced a premature stop codon in the chromosomal *lacZ* gene. In general, 250-500 cells were plated on LB-min+Xgal/IPTG (USB Biochemicals) agar plates. Frequency of allelic replacement was calculated by dividing the number of white colonies by the total number of colonies on plates. All phenotypic results are reported in Table S4-6.

Hierarchical Conjugation Assembly Genome Engineering (CAGE)

Donor and recipient strains were grown in 3 mL LB-min containing the appropriate positive selectable antibiotics. Once cells reached logarithmic-saturated growth, 2 mL samples of each culture were transferred to 2 mL Eppendorf tubes. Cells were washed three times in order to remove antibiotics present in the growth cultures. The washing procedure consisted of centrifuging samples at 5000 rpm for 2 minutes at room temperature, removing the supernatant, and re-suspending the cell pellet in fresh LB-min containing no antibiotics. After the final wash, the donor and recipient pellets were concentrated in 100 µL LB-min in order to enhance cell-cell contact during conjugation. Conjugation was initiated by combining 80 µL of ~20x concentrated donor culture and 20 µL of ~20x concentrated recipient culture. In order to minimize F pilus shearing, cells were gently mixed by pipetting. In order to minimize turbulence that can disrupt cell-cell contact during conjugation and to maximize genome transfer, the entire 100 µL donor-recipient mixture was transferred as a series of 2 x 20 µL and 6 x 10 µL spots onto an LB-min

agarose plate lacking antibiotics. This conjugation plate was incubated at 32° C for 0.5-2 hours, then the cells were re-suspended directly off of the plate using 1.5 mL LB-min and concentrated into a final volume of 250 µL. Desired recombinant genomes were selected by inoculating 5 µL of the concentrated post-conjugation culture into LB-min containing the correct combination of positive selection antibiotics (e.g., 10 µg/mL zeocin, 95 µg/mL spectinomycin, and 7.5 µg/mL gentamycin). The conjugated cells that populated the selected culture were then subjected to a negative selection using either *tolC* or *galK* to ensure proper DNA transfer of TAA codons at critical junction points between donor and recipient cells (see Figure 4-4).

This engineered conjugation method was tested for the first (1/32 genome, ~143 kb) and last (1/2 genome, ~2.3 Mb) chromosomal transfer steps in the hierarchical assembly experiment (Figure 4-1). By selecting for different combinations of markers across the donor and recipient genomes and subsequent screening of specific genomic loci, recombinant clones were isolated that contained the transfer of half or full (otherwise unmodified) genomes at a frequency of ~$2.5x10^{6}$ (from a population of 10^{9}-10^{10} cells), indicating the successful DNA transfer from an integrated *oriT* with episomal expression of conjugal factors. Equivalent frequencies were observed for full genome transfers.

Upon completion of the conjugation process, we also observed the anticipated loss of the *oriT-kan* cassette in the recombinant strain. This observation yields a subtle, yet very useful feature of our engineered conjugation system. By not inheriting the *oriT* sequence, the strains are positioned to proceed to a subsequent conjugation by a one-step integration of the *oriT-Kan* cassette in a new, targeted chromosomal locus (Figure 4-4A).

Illumina Whole Genome Sequencing

We prepared a paired-end Illumina sequencing library for three 1/8 genome strains C21 (regions 17-20), C22 (21-24), and C23 (25-28) using barcoded Illumina adapters. The barcoded library was sequenced on one lane using an Illumina GAII.

1. Genome prep (Qiagen Genome Prep kit)
2. Sheared 5 µg of gDNA to target size = 200 bp using covaris (estimated median band size 250 bp)
3. PCR purified DNA (QIAquick PCR purification kit)
4. End repair (Epicentre End-it™ DNA End-Repair kit)

Component	Volume (1x)
DNA sample	35
10x End-Repair Buffer	10
1 mM dNTPs	10
End-Repair Enzyme mix	5
dH2O	40
Total (µL)	100

5. Incubate at 25 C for 30 minutes
6. PCR purified DNA (QIAquick PCR purification kit)
7. A-tailing (NEB Klenow Fragment (3'->5' exo-))

Component	Volume (1x)
DNA sample	32
Klenow buffer	5
1 mM dATP	10
Klenow (3'->5' exo-)	3
Total (µL)	50

8. Incubate for 30 minutes at 37 C
9. PCR purified DNA (QIAquick PCR purification kit)
10. Adapter ligation (adapters complements of Morten Sommer)
    a. C21: TopPE-1 barcode = AGC
    b. C22.DO:T: TopPE-3 barcode = CTA
    c. C23: TopPE-4 barcode = TCT

Component	Volume (1x)
DNA sample	31
Rapid ligase buffer (2x)	35
PE adapter (50 µM)	2
Enzymatics Rapid (T4) ligase	2

11. Incubated at 20 C for exactly 10 minutes in a thermocycler, then immediately added PBI for the PCR purification
12. PCR purification (Qiagen MinElute PCR purification kit)
13. Gel purified adapter-ligated sequencing libraries on 2% agarose gel in 0.5x TBE (cut 2 mm bands corresponding to approximately 225 bp) (Qiagen Gel Purification kit)
14. PCR amplified sequencing libraries (2x KAPA HiFi Ready Mix; 11 cycles)
    a. Standard Illumina PE PCR Primers

Component	Volume (1x)
2X KAPA HiFi Ready Mix	25
PE_PCR-f	1
PE_PCR-r	1
dH2O	13
Template	4

Step	Temp	Time (min)
1	95	5:00
2	98	0:20
3	62	0:15
4	72	1:15
5	Go to step 2	11x
6	72	3:00
7	4	Forev

15. PCR purified (QIAquick PCR purification kit)
16. Validated sequencing library
    a. Cloned Illumina libraries using Invitrogen TOPO ZeroBlunt II
    b. Transformed into OneShot Top 10 electrocompetent cells
    c. Genewiz sequenced insert (Sanger sequencing; Seq. primer = M13 forward (Invitrogen): GTAAAACGACGGCCAG)
17. Size-selected sequencing libraries for ~225 bp bands (E-Gel® SizeSelect™ gels)
18. PCR purified Illumina libraries (Qiagen MinElute)
19. Quantitated contents of C21, C22.DO:T, and C23 libraries
    a. PAGE, Low DNA Mass Ladder (Invitrogen), and SYBR gold staining
    b. Densitometry
20. Prepared sequencing library by adding all 3 components to a final concentration of 10 nM
21. Sample QC, Clustering, and sequencing performed by BPF
    a. Standard Illumina PE Sequencing Primers

Genome Assembly and Sequence Analysis

Read Sorting and Processing

The raw Illumina reads in FASTQ format were preprocessed and sorted using the 6-bp barcodes in the paired end adaptors. Reads that contained anomalous barcodes were discarded. Reads containing any bases with a quality score of 2, also called the *Read Segment Quality Control Indicator (*based on *Illumina Quality Scores* by Tobia Man*)*, were discarded at this step, but all other reads were kept. After preprocessing, all reads were exactly 34 base pairs long.

Reference-based Assembly

The expected FASTA sequence of the EcNR2 parent strain was assembled by manually

modifying the FASTA sequence of E. coli K-12 strain MG1655 to reflect the removal of mutS and the insertion of the lambda prophage genome into the bioAB operon. Next, the preprocessed reads were sorted into separate files by pair group and the *Burrows-Wheeler Aligner* program (*BWA*) (*3*) was used to separately align the paired reads from each of the three strains to the expected EcNR2 FASTA sequence. The *sample* algorithm was used to align the reads. The distribution of insert sizes was inferred at runtime. During the read alignment step of *BWA*, (the *aln* command), a value of 10 was used for the suboptimal alignment cutoff.

Indel and SNP Filtering

After alignment, the *SAMtools* package (*4*) was used to create and sort BAM files for the assemblies. From these BAM files we generated a set of raw SNPs and short indels with respect to this reference assembly. These were then filtered using several criteria. First, using the *varFilter* script within *SAMtools*, we removed SNPs where the root mean squared mapping quality was less than 10, and indels where the root mean squared mapping quality was less than 25. We fitted the read coverage of each assembly to a gamma distribution and used the 99.95th and 0.05th percentile cutoffs for minimum and maximum read depth, beyond which SNPs and indels were discarded. We also discarded SNPs within 3 base pairs of a gap, and SNPs that occurred more densely than three within one 10 base pair window.

Region Masking

We used custom scripts to further filter SNPs and indels by masking regions of poor assembly. We masked regions containing many truncated reads, many incorrect read pairings, many non-unique alignments, and regions with motifs known to be problematic in Illumina sequencing (GGCnG). We defined truncated read regions as those containing multiple incompletely mapped reads, separated by less than one read length, containing at least 4

truncated reads and having a number of truncated reads totaling at least one half of the length of the contiguous region in which they were found.

Regions with incorrect read pairings were defined using the following method. We found read pairs whose insert size was outside of the 99.9th and 0.1th percentile of a fitted normal distribution of mate pair distance. These reads were counted in a 34-bp rolling window. As a thresholding step we chose contiguous regions where 10 or more of these reads were found in one window length. Additionally included were contiguous regions where only one read in a pair could be mapped, and these were thresholded with a rolling window in a similar fashion, using a 6-read cutoff. As a final masking step, we removed SNPs stemming from the replacement of amber stop codons as well as SNPs and indels where surrounding context was *GGCnC*, as these regions are known to be hotspots for Illumina sequencing errors.

Annotation

After removing SNPs and indels in the masked regions as described above, we attempted to associate the remaining SNPs and indels with functional consequences. We used a modified version of *Ensembl's SNP Effect Predictor* software (5), and the *Ensembl Bacteria* database to find SNPs that occurred within genes. We further categorized these by synonymous and non-synonymous coding changes, frameshift mutations, premature stop mutations, mutations in the 5' and 3' UTRs, and mutations less than 100 base pairs upstream of a transcript start site (Figure S4-8). Coordinates were lifted over from ECNR2 to MG1655 to permit annotation of the SNPs and indels. This resulted in C21, C22, and C23 having 4,5, and 5 mutations respectively having no corresponding liftover coordinates in ECNR2. These are referred to as the "unmappable" in Figure S4-8.

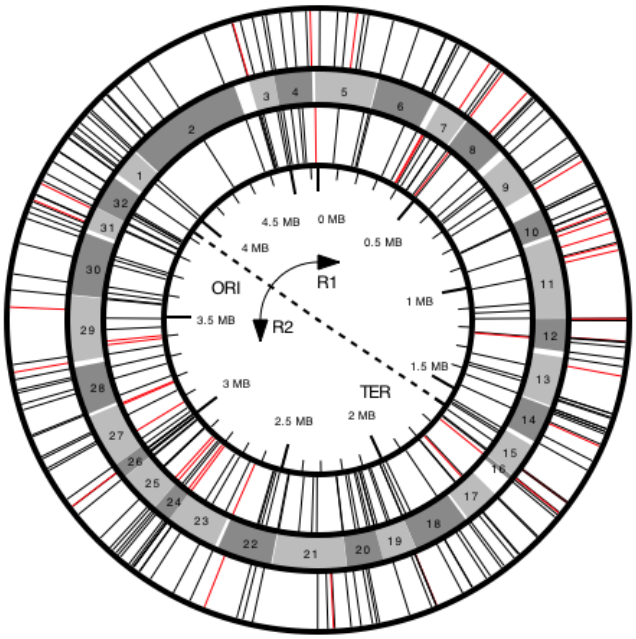


Figure S4-1. Circular representation of the *E. coli* genome depicting the coordinates and orientation of all TAG codons. TAG codons found in essential genes(*6*) are shown in red. The outer ring plots all clockwise transcribed TAG codons on the + DNA strand whereas the inner ring plots all counterclockwise transcribed TAG codons on the – DNA strand. The middle ring depicts the 32 sections of the genome targeted for TAG-to-TAA conversion. The inner circle plots the genomic coordinates, origin of replication (ORI), terminus (TER) and replichores 1 (R1) and 2 (R2).

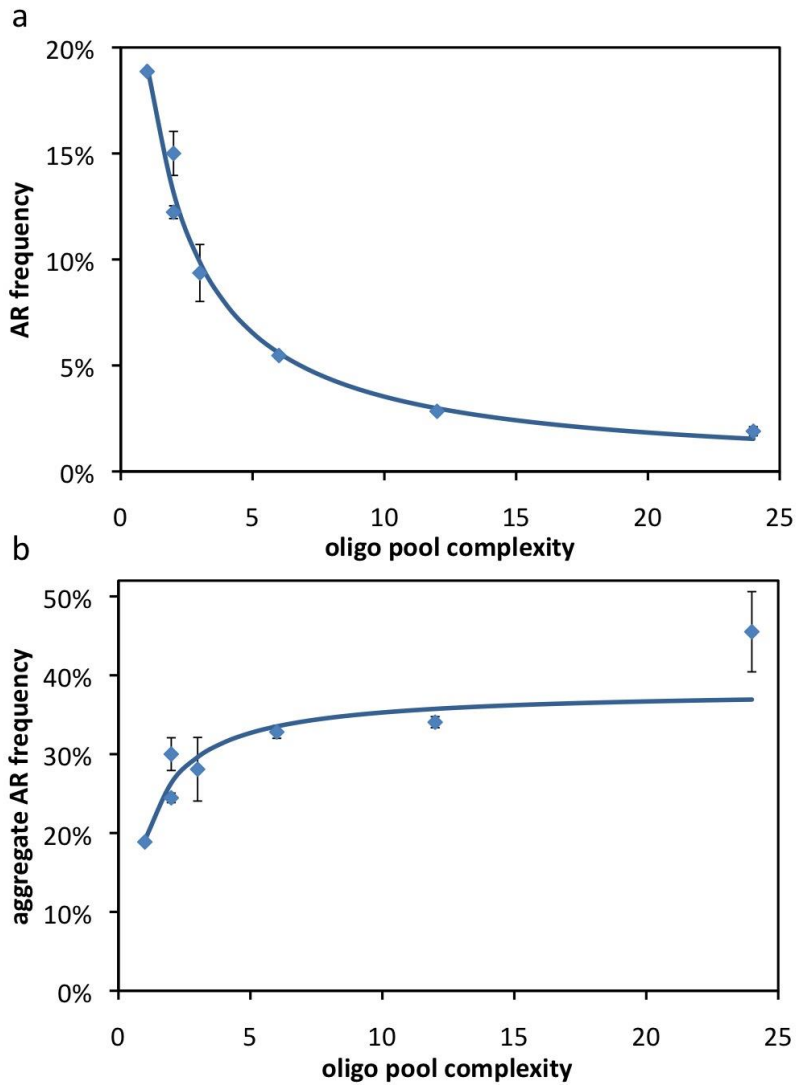


Figure S4-2. Multiplexing in MAGE leads to higher aggregate allele replacement (AR) frequencies. Multiplex oligo recombineering experiments were performed with equimolar oligo pools ranging in complexity from one to 24 oligos. AR frequencies were quantified for one conversion site corresponding to one oligo present in all pools. While individual AR frequencies (a) decrease as a function of higher complexity, the overall aggregate frequency (estimated as the product of individual frequency and pool complexity) (b) increases. Allele frequencies were measured using MASC-qPCR and curves are fit to the formula $y=a(1-e^{-b/x})$ for plot a and $y=ax(1-e^{-b/x})$ for plot b. Error bars indicate standard error (n=2).

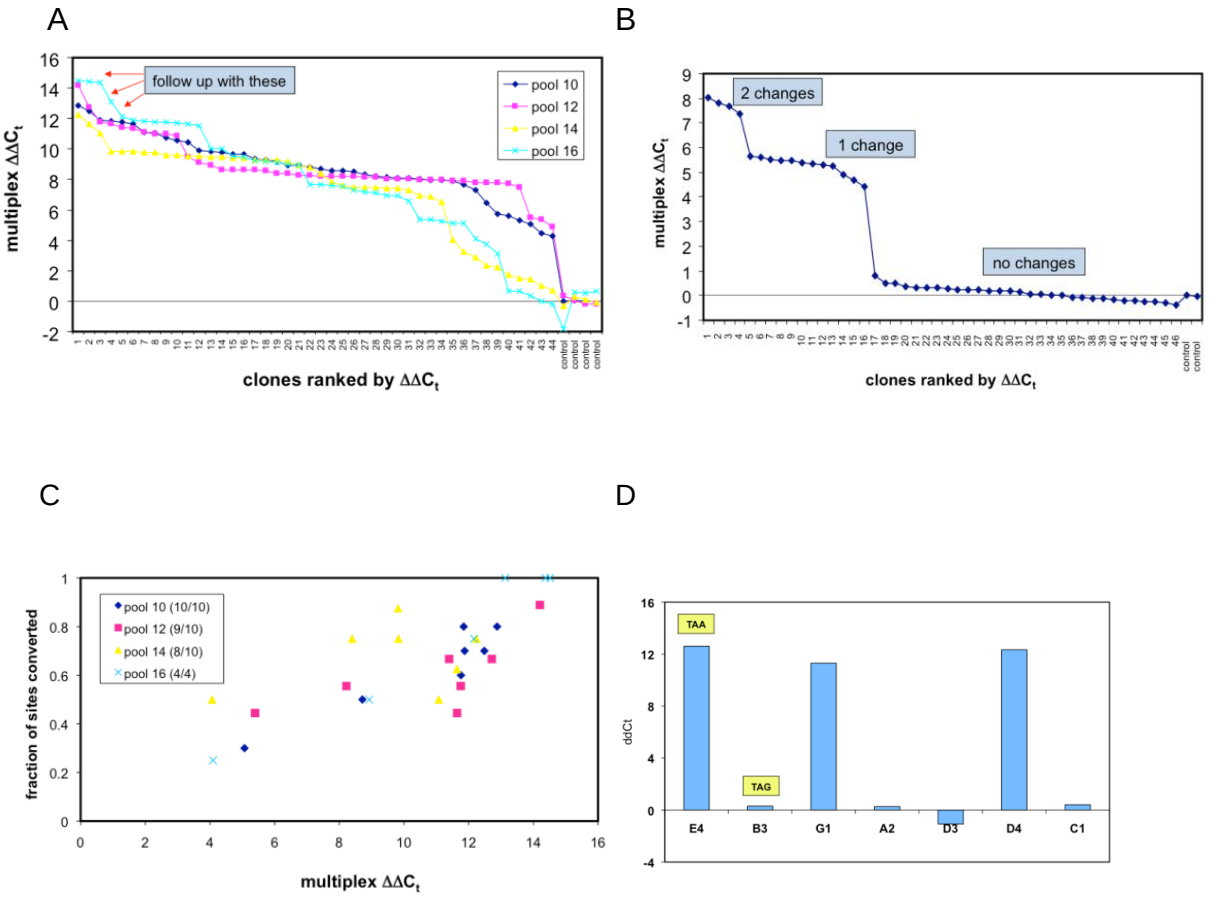


Figure S4-3. Multiplex allele-specific colony quantitative PCR (MASC-qPCR) to rapidly screen for the most modified clones. Multiplex reactions were used to determine which clones in the complex pool had been most highly modified. **A**. Clones from cultures receiving 18 cycles of MAGE processing (pools of 10 oligos) were sorted by their mutiplex $\Delta\Delta C_t$ scores. Small numbers of top-scoring clones (typically 3-5) were then assessed at each TAG site of interest. **B**. When only two modifying oligos are used for allele replacement, mutiplex $\Delta\Delta C_t$ values are more visibly clustered into groups representing 0, 1, or 2 modifications. **C**. Correlation between mutiplex $\Delta\Delta C_t$ scores from (A) and the number of specific modifications achieved. The top mutiplex $\Delta\Delta C_t$-scoring clone was found to have the most allele conversions roughly 70% of the time. The legend indicates the number of modifications observed in the top-scoring clones. **D**. Singleplex reactions were used to genotype the most promising clones. Shown are 7 clones assayed at the tfaS stop codon, with singleplex $\Delta\Delta C_t$ values of 0.0±0.7 for wild-type TAG and 12.1±0.7 for modified TAA.

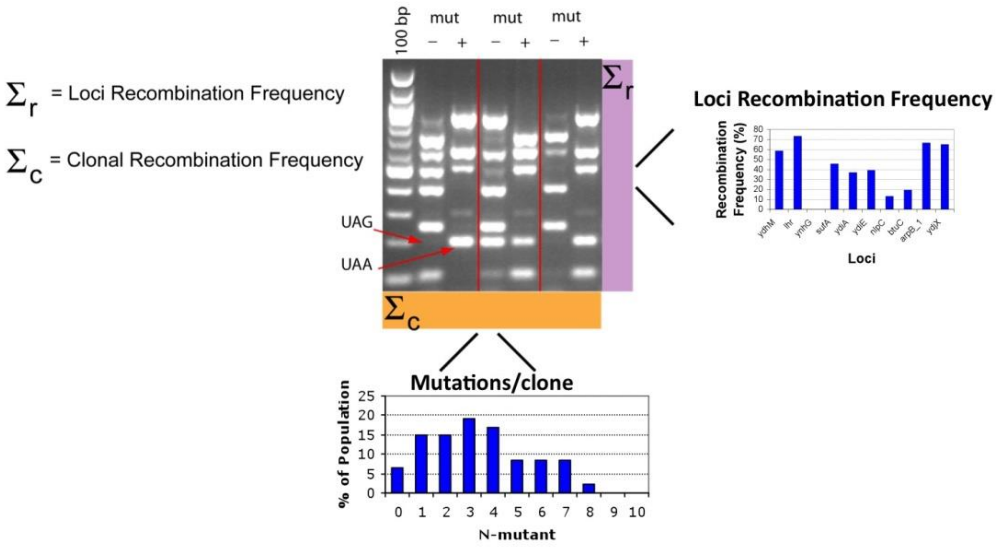


Figure S4-4. Multiplex allele-specific colony PCR (MASC-PCR) for the detection of codon conversions in each clone. The gel illustrates results from 3 MAGE-cycled clones. Ten sites are investigated in each lane where two MASC-PCRs are conducted for each clone: one reaction interrogates the TAG loci (-) and another reaction interrogates the TAA loci (+). Each reaction provides a binary output through the presence or absence of an amplicon band. Together, both TAG and TAA reactions provide sufficient information to determine the conversion status of a given codon. Summation of the rows of the 46 clones provides loci frequency data for each codon (plotted in Figure 2 in the main text). Summation of the columns of each clone provides a histogram of the mutations per clone (plotted in Figure 4-3 in the main text).

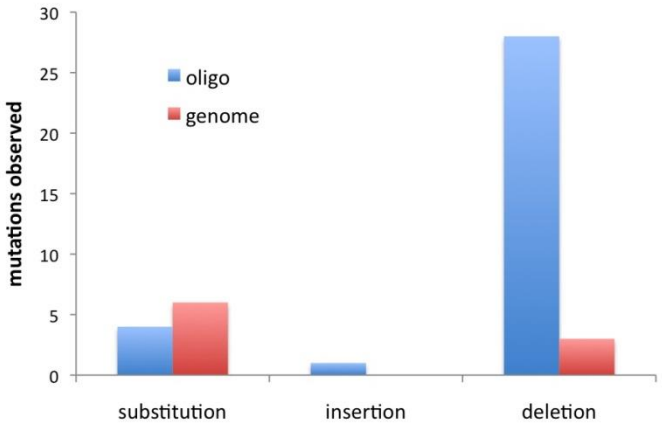


Figure S4-5. Off-target mutation in modified strains. From 96.3 kb of total Sanger sequencing, the majority of unwanted mutations were observed in regions corresponding to the annealing sites for the 90-mer oligonucleotides (blue). Sequenced mutations falling outside these 90 bp regions are shown in red. The principal error is a deletion, mostly single base deletions. These errors correspond to common defects arising from oligo synthesis.

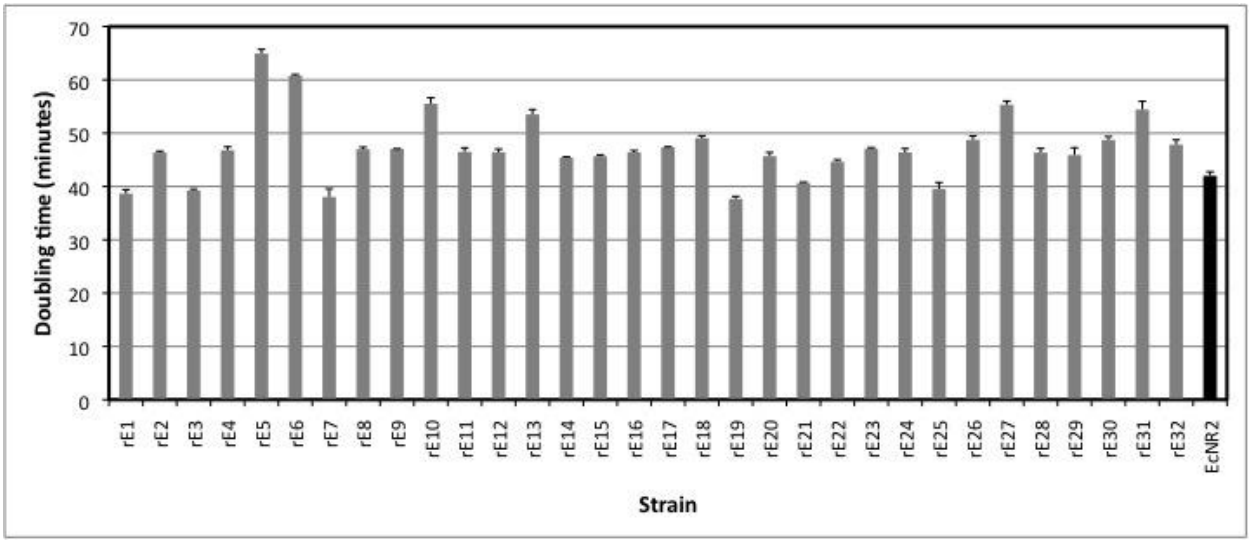


Figure S4-6. Individual growth rates of all 32 "top" recoded strains (10 out of 10 changes each, grey, rE1-rE32) following the successful replacement of all TAG-to-TAA codons versus that of the ancestral strain (black, EcNR2). A mix of increased and decreased growth rates was observed across the 32 strains with an average of 47 minutes/division. This is a mild decrease versus the growth rate (42 minutes/division) of the ancestral strain. Our parallelized MAGE approach across 32 strains allows us to easily identify strains with notable growth phenotypes (e.g., rE5, rE6). These strains can be investigated further to determine if these growth impediments are due to the codon changes or whether they arise from secondary mutations elsewhere in the genome.

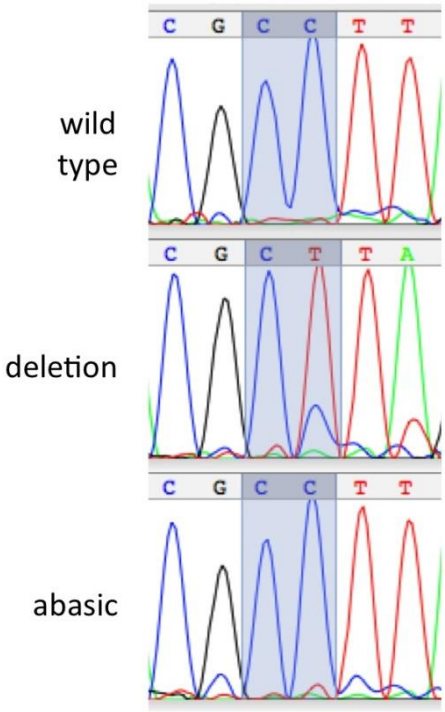
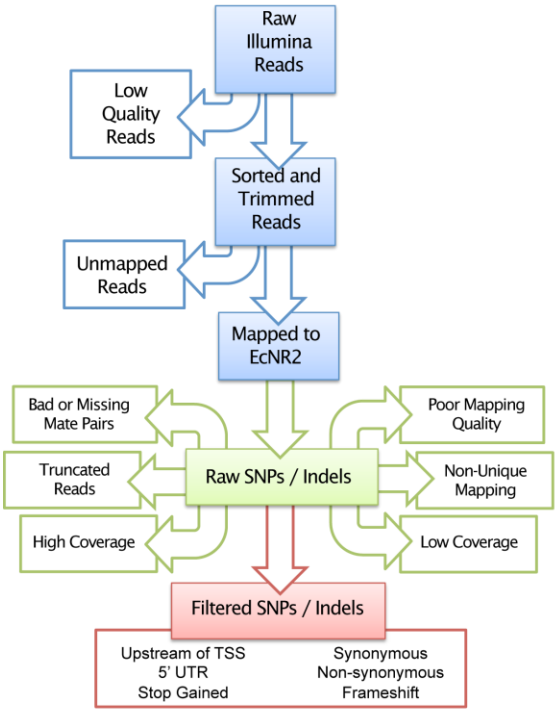


Figure S4-7. Some oligo defects can be fixed by host repair mechanisms. Oligos containing internal deletions are likely to result in equivalent mutations in the genome, but oligo synthesis chemistry can be optimized to minimize such deletions. In such cases, measures such as more aggresive deprotection and coupling conditions can then give rise to damaged oligos containing abasic sites. However, this second type of defect is readily repaired in the host. Three similar purified oligos were used to modify the selectable chromosomal tolC gene in separate cultures. An upstream modification (not shown) in each oligo creates a stop codon in tolC—selection against the tolC protein ensures only cells that have incorporated this oligo survive. PCR amplification of the resulting population and sequencing of this potentially heterogeneous product allows assessment of the effect of modifications at a second site. Top: only the initial stop codon was employed for this oligo, leaving the wild-type sequence C97-T102 of the tolC gene. Middle: this oligo coded for a deletion of C99, effectively shifting the subsequent peaks left one base position. However, a notable fraction (less than one-third) has not been shifted, indicating possible repair events (this fraction is very unlikely to arise from a defect in the oligo). Bottom: this oligo contained an abasic site at position 99, but the resulting population is almost completely wild-type, indicating likely cellular repair. These experiments were performed in strain EcNR2, which includes a deletion of the mutS mismatch repair gene.



	C21	C22	C23	Total
Raw Reads	-	-	-	50,050,024
Low Quality Reads	-	-	-	10,049,324
Sorted and Trimmed	11,753,698	16,353,342	11,893,660	40,000,700
Unmapped Reads	448,348	506,335	413,368	1,368,051
Mapped to EcNR2	11,305,350 (96.19%)	15,847,007 (96.90%)	11,480,292 (96.52%)	38,632,64 (96.58%)
Raw SNPs / Indels	269	323	227	819
Poor Mapping Quality	108	148	75	331
Bad/Missing Mates	11	11	10	32
Truncated Reads	4	10	7	21
High Coverage	2	0	2	4
Low Coverage	3	5	3	11
Non-Unique Mapping	7	10	10	27
Filtered SNPs / Indels	110	128	102	340
Synonymous	29	33	24	86
Non-synonymous	50	46	41	137
Frameshift	11	11	10	32
Stop gained	2	4	2	8
5' UTR	2	3	3	8
Upstream of Transcript	2	4	1	7
Within Noncoding	2	8	4	14
Unmappable	4	5	5	14
No annotation	8	14	12	34

Figure S4-8. Bioinformatics sequencing analysis process and secondary mutation breakdown. We sequenced the entire genomes of three 1/8 recoded strains and identified off-target SNPs and short (1 bp) indels using BWA sequencing alignment (*3*). On-target TAG conversions are not included in this analysis. We found an average of 113 mutations/genome after each strain went through approximately 960 doublings, multiple lambda red inductions, and several conjugations. This corresponds to 2E-8 mutations/bp/doubling, which is consistent with the predicted basal mutation rate of the ancestral strain (EcNR2). These results indicate that MAGE and CAGE do not significantly compromise genome stability. Also see Figure S4-9 and Tables S4-8 to S4-11 for supporting data and information.

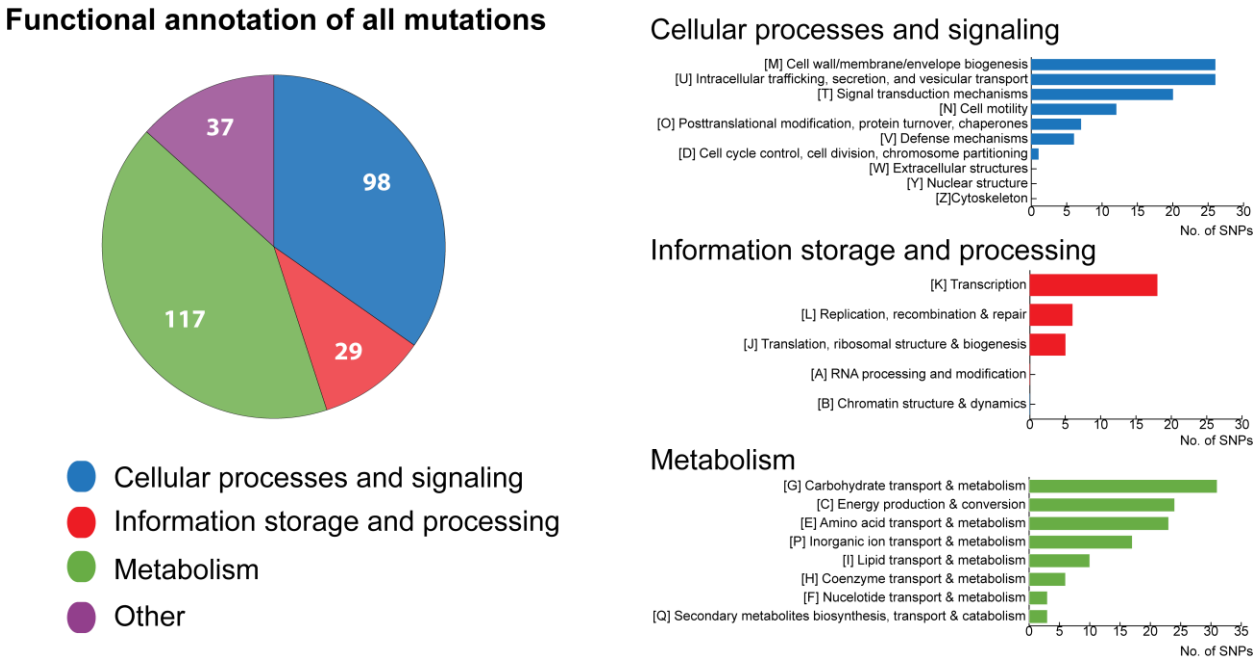


Figure S4-9 Functional annotation of all mutations as indicated by the COG category of the gene or regulatory region associated with the SNP or indel. Functional annotation of all mutations are summed across the three sequenced genomes. See Table S4-8 for complete list of secondary mutations.

Supplemental Tables can be found at
<http://www.sciencemag.org/content/suppl/2011/07/13/333.6040.348.DC1/Isaacs.SOM.pdf>

References (Supporting Online Material)

1. D. Yu *et al.*, *Proceedings of the National Academy of Sciences of the United States of America* **97**, 5978 (May 23, 2000).
2. H. H. Wang *et al.*, *Nature* **460**, 894 (Aug 13, 2009).
3. H. Li, R. Durbin, *Bioinformatics* **25**, 1754 (Jul 15, 2009).
4. H. Li *et al.*, *Bioinformatics* **25**, 2078 (Aug 15, 2009).
5. W. McLaren *et al.*, *Bioinformatics* **26**, 2069 (Aug 15, 2010).
6. S. Y. Gerdes *et al.*, *J Bacteriol* **185**, 5673 (Oct, 2003).

APPENDIX D

Supplemental Material for Genomically Recoded Organisms Impart New Biological Functions

This supplemental material is reproduced with permission from its initial publication:

Lajoie MJ, Rovner AJ, Goodman DB, Aerni HR, Haimovich AD, Kuznetsov G, Mercer JA, Wang HH, Carr PA, Mosberg JA, Rohland N, Schultz PG, Jacobson JM, Rinehart J, Church GM, Isaacs FI (2013) *Genomically Recoded Organisms Impart New Biological Functions*. **Science** 342: 357-60.

Tables and Figures have been renamed to be consistent with CHAPTER 5.

Table of contents for Supplemental Online Text

A. Materials and Methods pages 227-239
B. Time and cost pages 240-242
C. Construction of a recoded genome pages 243-251
D. GRO nomenclature and applications pages 252-253
E. Partial recoding strategies for reassigning UAG codon function pages 254-255
F. Analysis of MAGE and CAGE pages 256-258
G. Analysis of recoded lineage pages 259-271
H. Mass spectrometry pages 272-276
I. NSAA incorporation pages 277-285
J. Increased T7 resistance pages 286-292
K. Selectable markers used in this study pages 293-296
L. References pages 297-299

Supplemental Figures

Figure S5-1. Properties of genomically recoded organisms (GROs) page 253
Figure S5-2. Fully recoded strain cell morphology in the presence or absence of RF1 page 260
Figure S5-3. Construction and analysis of C321.ΔA page 263
Figure S5-4. Doubling times in recoded strains +/- RF1 page 277
Figure S5-5. C0.B*.ΔA::S outgrowth is impaired following overnight pAcF and pAzF expression page 279
Figure S5-6. Complete removal of all native UAGs permits robust NSAA incorporation page 281
Figure S5-7. Extracted ion chromatograms are shown for pAcF incorporation into the YgaU peptide page 273
Figure S5-8. Bacteriophage T7 plaques on recoded host strains page 287
Figure S5-9. T7 kinetic lysis curves (MOI = 5) page 289
Figure S5-10. One step growth curve averaged across replicates 3-6 page 292
Figure S5-11. Putative Conj20 rearrangement causing tolC mistargeting page 245
Figure S5-12. Strategic marker placement allowed removal of undesired structural variant from Conj26 page 246
Figure S5-13. Strain Conj30 was prepared by two serial conjugations page 247
Figure S5-14. Redundant recoding for Conjugation 31 page 248
Figure S5-15. MAGE oligo simultaneously converting UAGs of convergently overlapping yegV and yegW genes page 249
Figure S5-16. Distribution of the number of amino acids added to the C-terminus of genes as a result of UAG read-through page 254
Figure S5-17. Native UAGs cause detrimental pleiotropic effects after codon reassignment page 278
Figure S5-18. Western blots of GFP variants in the soluble/insoluble fractions page 284
Figure S5-19. Western blots of GFP variants in a crude lysate page 285
Figure S5-20. Plaque area raw data page 288
Figure S5-21. One step growth curves were performed using hosts C321 and C321.ΔA in six replicates page 291
Figure S5-22. One step growth curve parameters page 292

Supplemental Tables: (‡ supp tables accessible at <www.sciencemag.org/content/342/6156/357/suppl/DC1>)

Table S5-1. Doubling times and Max OD_{600} of recoded genome lineage pages 261-262
Table S5-2. Total estimated number of doublings required to reassign UAG page 258
Table S5-3‡. Summaries of mutations observed in each strain of recoding lineage separate file
Table S5-4‡. All mutations observed in recoded strain lineage separate file
Table S5-5. Essential and important genes terminating with UAG page 255
Table S5-6. Summary of survey proteomic analysis of strains page 273
Table S5-7. Summary of in-depth proteomics of strains page 273
Table S5-8. Summary of identified pAcF containing peptides page 274
Table S5-9. Summary of all identified proteins with pAcF incorporation at UAG codon(s) page 274
Table S5-10. Summary from the proteomic analysis of the TiO_2 enriched fraction of strains containing Sep-TECH page 275
Table S5-11. Summary of Sep-containing peptides identified by proteomics from two biological replicates each page 276
Table S5-12. Summary of all identified proteins with Sep incorporation at an amber stop codon page 276
Table S5-13. LC-MS/MS of C13.ΔA::S after appearance of natural suppression page 282
Table S5-14. Pairwise statistical comparison of plaque areas page 288
Table S5-15. One-step growth parameters: eclipse time, rise rate, and burst size page 292
Table S5-16‡. UAG codons converted to UAA codons in each strain of recoding lineage separate file
Table S5-17. Time required to reassign UAG page 240
Table S5-18. DNA cost for reassigning the UAG codon page 240
Table S5-19‡. DNA oligonucleotides used in this study separate file
Table S5-20. Positions of markers for CAGE and window sizes for conjugal junctions pages 241-242
Table S5-21. UAG codons that were retained in Conj31 after CAGE pages 249
Table S5-22. UAG codons that were not targeted in the original design page 250
Table S5-23. UAG codons found in genes re-annotated as phantom page 250
Table S5-24. Summary of snpEFF types pages 266
Table S5-25. Summary of blastn results for potential MAGE oligo mistargeting regions page 268-269
Table S5-26‡. Putative MAGE oligo mistargeting events separate file
Table S5-27‡. All uncharacterized Pindel breakpoint events separate file
Table S5-28‡. All complete Pindel structural events separate file
Table S5-29‡. All high quality Breakdancer events separate file
Table S5-30. UAG terminating genes in bacteriophages T4 and T7 page 286-287
Table S5-31‡. Sequencing coverage of genome in each strain of recoding lineage separate file
Table S5-32‡. Summary of CAGE lineage for removing all instances of UAG from a single genome separate file
Table S5-33. Sequences of GFP variants containing UAG codons page 296
Table S5-34‡. List of all 321 targeted UAG locations in MG1655 separate file
Table S5-35‡. List of cassette insertions and structural events used to generate C321.ΔA Genbank annotation separate file
Table S5-36‡. List of variants used to generate C321.ΔA strain separate file
Table S5-37. Recoded strains and their genotypes page 252

A. Materials and Methods

All DNA oligonucleotides were purchased with standard purification and desalting from Integrated DNA Technologies (Table S5-19). Unless otherwise stated, all cultures were grown in LB-Lennox medium (LB^L, 10 g/L bacto tryptone, 5 g/L sodium chloride, 5 g/L yeast extract) with pH adjusted to 7.45 using 10 M NaOH. LB^L agar plates were LB^L plus 15 g/L bacto agar. Top agar was LB^L plus 7.5 g/L bacto agar. MacConkey agar was prepared using BD Difco™ MacConkey agar base according to the manufacturer's protocols. M9 medium (6 g/L Na_2HPO_4, 3 g/L KH_2PO_4, 1 g/L NH_4Cl, 0.5 g/L NaCl, 3 mg/L $CaCl_2$) and M63 medium (2 g/L $(NH_4)_2SO_4$, 13.6 g KH_2PO_4, 0.5 mg $FeSO_4 \cdot 7H_2O$) were adjusted to pH 7 with 10 M NaOH and KOH, respectively. Both minimal media were supplemented with 1 mM $MgSO_4 \cdot 7H_2O$, 0.083 nM thiamine, 0.25 μg/L D-biotin, and 0.2% w/v carbon source (galactose, glycerol, or glucose).

The following selective agents were used: carbenicillin (50 μg/mL), chloramphenicol (20 μg /mL), kanamycin (30 μg/mL), spectinomycin (95 μg/mL), tetracycline (12 μg/mL), zeocin (10 μg/mL), gentamycin (5 μg/mL), SDS (0.005% w/v), Colicin E1 (ColE1; ~10 μg/mL), and 2-deoxygalactose (2-DOG; 0.2%). ColE1 was expressed in strain JC411 and purified as previously described (*26*). All other selective agents were obtained commercially.

The following inducers were used at the specified concentrations unless otherwise indicated: anhydrotetracycline (30 ng/μL), L-arabinose (0.2% w/v).

p-acetyl-L-phenylalanine (pAcF) was purchased from PepTech (# AL624-2) and used at a final concentration of 1 mM. O-phospho-L-serine (Sep) was purchased from Sigma Aldrich (# P0878-25G) and used at a final concentration of 2 mM.

Strains

All strains were based on EcNR2 (*11*) (*Escherichia coli* MG1655 Δ*mutS*::*cat* Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*]). Strains C321 [strain 48999 (www.addgene.org/48999)] and C321.ΔA [strain 48998 (www.addgene.org/48998)] are available from addgene.

Selectable marker preparation

Selectable markers were prepared using primers described in Table S5-19. PCR reactions (50 μL per reaction) were performed using Kapa HiFi HotStart ReadyMix according to the manufacturer's protocols with annealing at 62 °C. PCR products were purified using the Qiagen PCR purification kit, eluted in 30 μL of dH_2O, quantitated using a NanoDrop™ ND1000 spectrophotometer, and analyzed on a 1% agarose gel with ethidium bromide staining to confirm that the expected band was present and pure.

MAGE and λ Red-mediated recombination

MAGE (*13*), CoS-MAGE (*14*), and λ Red-mediated recombination (*27*) were performed as previously described. Briefly, an overnight culture was diluted 100-fold into 3 mL LB^L plus antibiotics and grown at 30 °C in a rotator drum until mid-log growth was achieved (OD_{600} ~0.4-0.6). Lambda Red was induced in a shaking water bath (42 °C, 300 rpm, 15 minutes), then induced culture tubes were cooled rapidly in an ice slurry for at least two minutes. Electrocompetent cells were prepared at 4 °C by pelleting 1 mL of culture (centrifuge at 16,000

rcf for 20 seconds) and washing the cell pellet twice with 1 mL ice cold deionized water (dH_2O). Electrocompetent pellets were resuspended in 50 µL of dH_2O containing the desired DNA. For MAGE oligos, no more than 5 µM (0.5 µM of each oligo) was used. For CoS-MAGE, no more than 5.5 µM (0.5 µM of each oligo including the co-selection oligo) was used. For dsDNA PCR products, 50 ng was used. Cells were transferred to 0.1 cm cuvettes, electroporated (BioRad GenePulser™, 1.78 kV, 200 Ω, 25 µF), and then immediately resuspended in 3 mL LB^L (MAGE and CoS-MAGE) or 1.5 mL LB^L (dsDNA). Recovery cultures were grown at 30 °C in a rotator drum. For continued MAGE cycling, cultures were recovered to mid-log phase before being induced for the next cycle. To isolate monoclonal colonies, cultures were recovered for at least 3 hours (MAGE and CoS-MAGE) or 1 hour (dsDNA) before plating on selective media. For *tolC* and *galK* negative selections, cultures were recovered for at least 7 hours to allow complete protein turnover before exposure to ColE1 and 2-deoxygalactose, respectively.

CAGE

CAGE was performed as previously described (*11*). Briefly, conjugants were grown to late-log phase in all relevant antibiotics (including tetracycline in the donor culture to select for the presence of conjugal plasmid pRK24 (*28*)). At mid-log growth, 2 mL of each culture was transferred to a 2 mL microcentrifuge tube and pelleted (5000 rcf, 5 minutes). Cultures were washed twice with LB^L to remove antibiotics, then the pellets were resuspended in 100 µL LB^L. Donor (10 µL) and recipient (90 µL) samples were mixed by gentle pipetting and then spotted onto a pre-warmed LB^L agar plate (6 x 10 µL and 2 x 20 µL spots). Conjugation proceeded at 30 °C without agitation for 1 – 24 hours. Conjugated cells were resuspended off of the LB^L agar plate using 750 µL liquid LB^L, and then 3 µL of the resuspended conjugation was inoculated into 3 mL of liquid LB^L containing the appropriate selective agents. The population with the correct resistance phenotype was then subjected to ColE1 negative selection to eliminate cells that retained *tolC*.

Each round of conjugation, genotyping, and strain manipulation required a minimum of 5 days to complete. On day 1, the conjugation and positive selections were performed. On day 2, the population of cells exhibiting the desired resistance phenotype was subjected to a ColE1 selection to eliminate candidates that retained *tolC*. The ColE1-resistant population was then spread onto plates to isolate monoclonal colonies. On day 3, candidate colonies were grown in a 96-well format and screened for the desired genotypes via PCR (to confirm loss of *tolC*) and MASC-PCR (to confirm the presence of the desired codon replacements). On day 4, *tolC* or *kanR*-oriT was recombined directly into one of the positive markers, and recombinants were plated on LB^L plates containing SDS or kanamycin, respectively. On day 5, candidate colonies were grown in liquid LB^L containing SDS or kanamycin and used as PCR template to confirm successful replacement of positive selection markers with *tolC* or *kanR*-oriT. These strains were ready for the next conjugation.

Positive/Negative selections

Positive selection for tolC: TolC provides robust resistance to SDS (0.005% w/v) in LB^L (both liquid and LB^L agar).

Negative selection for tolC: After *tolC* was removed via λ Red-mediated recombination or conjugation, cultures were recovered for at least 7 hours prior to ColE1 selection. This was

enough time for the recombination to proceed and for complete protein turnover in the recombinants (*i.e.* residual TolC protein no longer present). ColE1 selections were performed as previously described (*11*). Briefly, pre-selection cultures were grown to mid-log phase (OD_{600} ~0.4), then diluted 100-fold into 150 μL of LB^L and LB^L + ColE1. Once growth was detected, monoclonal colonies were isolated on non-selective plates and PCR screened to confirm the loss of *tolC*.

Positive selection for galK: GalK is necessary for growth on galactose (0.2% w/v) as a sole carbon source. It is important to thoroughly wash the cells with M9 media to remove residual carbon sources prior to selection in M63 + galactose (both liquid and M63 agar). Noble agar must be used, since Bacto agar may contain contaminants that can be used as alternative carbon sources.

Negative selection for galK: After *galK* was removed via λ Red-mediated recombination or conjugation, cultures were recovered for at least 7 hours prior to 2-DOG selection. This was enough time for the recombination to proceed and for complete protein turnover in the recombinants (*i.e.* residual GalK protein no longer present). 2-DOG selections were performed as previously described (*29*). Briefly, pre-selection cultures were grown to mid-log phase (OD_{600} ~0.4), washed three times in M9 medium to remove residual nutrients from LB^L, and then inoculated into M63 + 0.2% glycerol and M63 + 0.2% glycerol + 0.2% 2-DOG. Once growth was detected, monoclonal colonies were isolated on non-selective plates (LB^L agar or MacConkey agar) and PCR screened to confirm the loss of *galK*. When possible, colonies were streaked onto MacConkey + 0.2% galactose indicator plates (white colonies are Gal- and red colonies are Gal+) prior to PCR screening, but MacConkey media is toxic to strains that do not express TolC, which provides resistance to bile salts. We also found that 2-DOG selection was effective in LB^L, but PCR screening was important because LB^L + 2-DOG selection was less stringent.

Screening for galK and malK: Cultures were diluted and plated for single colonies on MacConkey agar + 0.2% galactose (galK) or MacConkey agar + 0.2% maltose (malK) indicator plates (white colonies are Gal- or Mal-, and red colonies are Gal+ or Mal+). The genotypes were confirmed *via* PCR.

Genotyping

After λ Red-mediated recombination or conjugation, colony PCR was used to confirm the presence or absence of selectable markers at desired positions. Colony PCR (10 μL per reaction) was performed using Kapa 2G Fast HotStart ReadyMix according to the manufacturer's protocols with annealing at 56 °C. Results were analyzed on a 1% agarose gel with ethidium bromide staining.

Multiplex allele-specific colony PCR (MASC-PCR) was used to simultaneously detect up to 10 UAG→UAA conversions as previously described (*11*). Briefly, each allele was interrogated by two separate PCRs to detect the UAG/UAA status. The two reactions shared the same reverse primer but used different forward primers whose 3′ ends annealed to the SNP being assayed. Amplification only in the wt-detecting PCR indicated a UAG allele, whereas amplification only in the mutant-detecting PCR indicated a UAA allele. Each primer set produced a unique

amplicon size corresponding to its target allele (100, 150, 200, 250, 300, 400, 500, 600, 700 and 850 bp). Template was prepared by growing monoclonal colonies to late-log phase in 150 µl LB^L and then diluting 2 µl of culture into 100 µl dH_2O. Initially, we used Qiagen Multiplex PCR kit, but KAPA 2G Fast Multiplex Ready Mix produced cleaner, more even amplification across our target amplicon size ranges. Therefore, typical MASC-PCR reactions contained KAPA 2G Fast Multiplex ReadyMix (Kapa Biosystems, # KK5802) and 10X Kapa dye in a final volume of 10 µl, including 2 µl of template and 0.2 µM of each primer. PCR activation occurred at 95°C (3 min), followed by 27 cycles of 95°C (15 sec), 63–67°C (30 sec; annealing temperature was optimized for each set of MASC-PCR primers), and 72°C (70 sec). The final extension was at 72°C (5 min). MASC-PCR results were analyzed on 1.5% agarose gels with ethidium bromide staining to ensure adequate band resolution.

Sanger sequencing was performed by Genewiz or Eton Bioscience, Inc.

Genomic DNA for whole genome sequencing was prepared using a Qiagen Genomic DNA purification kit or by simultaneously lysing raw culture and shearing genomic DNA using a Covaris E210 AFA Ultrasonication machine. Illumina libraries were prepared as previously described (*30*). Each strain was barcoded with a unique 6 bp barcode for pooling. Up to 16 strains were pooled for sequencing on a single HiSeq lane, and up to 4 genomes were pooled for sequencing on a single MiSeq lane. Whole genome sequencing was performed using Illumina HiSeq or MiSeq systems. The HiSeq samples were sequenced with paired end 50 bp or 100 bp reads, and the MiSeq samples were sequenced with paired end 150 bp reads.

Sequencing analysis

In order to analyze the sequencing data from 68 distinct genomes, we developed a software pipeline that connects several modular tools and custom scripts for analysis and visualization. The goal of our pipeline was to identify SNPs and structural variants relative to the reference genome *E. coli* K-12 MG1655 (U00096.2, GI:48994873). Note that we use the term SNP to mean any small mismatches or indels identified by Freebayes (<22 bp). We use the term structural variant to refer to large insertions detected by Breakdancer and Pindel, deletions, or other significant junction events (confirmed variants of size 170 bp and 776 bp in C321.ΔA).

FASTQ conversion to SAM/BAM: FASTQ reads were split using individual genome barcodes with the FASTX toolkit (*31*). After splitting and trimming of the 6 bp barcode, FASTQ files for individual reads were aligned to the reference genome (*E. coli* K-12 MG1655 or the C321.ΔA predicted genome sequence) using Bowtie2 version 2.0.0-beta5 (*32*) with local alignment and soft-clipping enabled. PCR duplicates were removed using the Picard toolkit <http://picard.sourceforge.net/> and reads were realigned around short indels using the Genome Analysis Toolkit (*33*).

SNP calling using Freebayes: SNPs were called using the Freebayes package (arXiv:1207.3907v2 [q-bio.GN]). SNP calls were made using a *--ploidy* flag value of 2, in order to catch SNPs that occur in duplicated regions. These SNPs show up as heterozygous calls in the output. The minimum alternate fraction for such calls was set at 0.4. The p-value cutoff was set at 0.001. SNPs from all genomes were called simultaneously, using the *--no-ewens-priors* and *--no-marginals* flags. The *--variant-input flag was used to provide Freebayes with the recoded*

SNP (UAG-to-UAA) positions as putative variants to call regardless of evidence. Reads supporting SNPs were required to have a minimum mapping quality of 10 and a minimum base quality of 30. Mapping quality was not otherwise used to assess SNP likelihoods (*--use-mapping-quality* was disabled). We ran Freebayes as described above to generate a single VCF file containing all variants for all samples. This VCF file was then further analyzed and filtered before as described below, before generating the summarizing diagram Figure S5-3.

SNP Effect using snpEFF: SnpEff 2.0.5d (*34*) was used to annotate variants and to predict effects for called SNPs. First, the reference genome's annotated GenBank Record (GI:48994873) was used to create a genome database, and the VCF records were annotated for coding effects only.

Final SNP filtering: In addition to the Freebayes SNP identification criteria, we used additional metrics to filter out SNPs that could not be called with high confidence. This additional filtering helped to reduce the complexity of the relationship of variants across all sequenced genomes in order to plot Figure S5-3. Note that this filtering resulted in some low-evidence variants being temporarily ignored in the aggregate analysis. However, these were carefully triaged and identified in the process of generating the sequence annotation file for the final C321.ΔA strain.

v. All 'heterozygous' calls were filtered out, as these represent SNPs whose reads map to multiple locations in the genome.
vi. SNPs that were present in fewer than three samples and could not be called either present or absent in >20 strains due to poor coverage or read mapping quality were filtered out.
vii. SNPs were filtered out if they were covered by ≤ 20 reads with good mapping quality across all genomes.
viii. SNPs that could be called absent or present in fewer than three genomes were removed.

Structural variants using Pindel and Breakdancer: Pindel (*35*) and Breakdancer (*36*) were both used to find potential structural variants in the genomes. First, Picard <http://picard.sourceforge.net/> was used to gather insert size metrics per genome. This information, along with the aligned BAM data, was run through Pindel. The Pindel output was converted to VCF using the *pindel2vcf* tool. We required at least 20 reads to support a breakpoint or junction. The *breakdancer_max* program in Breakdancer was also used to find structural variants. For Breakdancer, at least 8 read pairs were required to support a called structural event.

We manually corroborated structural variant calls from Pindel and Breakdancer through visual examination of read alignments. Since we observed a high-rate of false-positive and false-negative calls with these toolswe did not include them in our final strain analysis in the main text. Still, the Pindel and Breakdancer data were useful in troubleshooting cassette insertions and intentional gene knockouts and replacements.

Future work to combine evidence from these and additional tools might lead to a more robust, comprehensive, and high throughput method to validate structural variants using only short-read sequencing data.

Breakdancer predicted 49 unique events, and 187 total events across 69 strains. Because Breakdancer cannot call across multiple strains simultaneously and only gives approximate event locations based on read-pair distances, events that occurred in multiple samples were identified by using similar event start and end locations. Breakdancer predicted a total of 21 unique deletions, 5 unique inversions, and 23 unique translocations.

Pindel used split read data to predict both uncharacterized breakpoints and whole structural events. 258 unique uncharacterized breakpoints were found; 230 of these occur in only a single sample. Pindel also predicted 79 unique structural events. 9 were large deletions, 59 were insertions of unknown size, and 11 were inversions.

Coverage analysis: Coverage for each genome was analyzed using the bedtools (*37*) programs *makewindows* and *multicov.* The genome was split into 50 bp windows and BAM coverage was assessed for each window. A custom python script was used to take this information and find contiguous windows of low and high coverage, indicative of gene amplifications and deletions. These results are included as supplemental Table S5-31.

Confirming cassette insertion sites: Known insertion sites of CAGE antibiotic resistance markers were confirmed by selecting the reads that were soft clipped and/or not aligned to the MG1655, and aligning them to the known cassette sequences using Bowtie. Cassette insertion locations were inferred using the alignment locations of paired reads in which one read mapped to a cassette and the other mapped to a location on the genome.

Visually confirming SNPs and structural variants: The *tview* tool in the *Samtools* package (*38*) was used to visually inspect individual UAG SNPs and to assess the veracity of low-confidence SNP and structural variant calls.

Generating genome figures: Figure S5-3 was created using custom software written in R and Processing.

Fitness analysis

To assess fitness, strains were grown in flat-bottom 96-well plates (150 μL LB^L, 34 °C, 300 rpm). Kinetic growth (OD_{600}) was monitored on a Biotek H4 plate reader at 5 minute intervals. Doubling times were calculated by $t_{double} = c*\ln(2)/m$, where c = 5 minutes per time point and m is the maximum slope of $\ln(OD_{600})$. Since some strains achieved lower maximum cell densities, slope was calculated based on the linear regression of $\ln(OD_{600})$ through 5 contiguous time points (20 minutes) rather than between two pre-determined OD_{600} values. To monitor fitness changes in the CAGE lineage, growth curves were measured in triplicate, and their average was reported in Figure 5-2 and Table S5-1. To determine the effect of RF1 removal and NSAA incorporation on the panel of recoded strains (Table 1), growth curves were measured in triplicate (Figure 5-3A, Figure S5-8). Statistics were based on a Kruskal-Wallis one-way ANOVA followed by Dunn's multiple comparison test, where $*p < 0.05$, $**p < 0.01$, and $***p < 0.001$.

To assess re-growth phenotypes from long-term NSAA expression, overnight cultures were first grown in LB^L supplemented with chloramphenicol to maintain the pEVOL plasmids. These cultures were passaged into LB^L containing chloramphenicol, arabinose (to induce pEVOL), and either pAcF or pAzF depending on whether pEVOL-pAcF or pEVOL-pCNF was used. Growth with shaking at 34°C was monitored using a Biotek H1 or a Biotek Eon plate reader with OD_{600} readings every 10 minutes (pAcF) or 5 minutes (pAzF). After 16 hours of growth, the expression cultures were passaged into identical expression conditions and the growth curves were monitored with the same protocols.

NSAA incorporation assays

Plasmids and strains for NSAA incorporation: p-acetyl-L-phenylalanine (pAcF) incorporation was achieved using pEVOL-pAcF (*9*) which contains two copies of pAcF-RS and one copy of $tRNA_{CUA}^{opt}$. The pEVOL-pAcF plasmid was maintained using chloramphenicol resistance. One copy of pAcF-RS and $tRNA_{CUA}^{opt}$ were constitutively expressed, and the second copy of pAcF-RS was under araBAD-inducible control (0.2% L-arabinose).

O-phospho-L-serine (Sep) incorporation was achieved by expression of $tRNA^{Sep}$ from pSepT and both EFSep (EF-Tu variant capable of incorporating Sep) and SepRS from pKD-SepRS-EFSep (*21*). To prevent enzymatic dephosphorylation of Sep *in vivo*, the gene encoding phosphoserine phosphatase (*serB*), which catalyzes the last step in serine biosynthesis, was inactivated. Specifically, Glu93 (GAA) was mutated to a premature UAA stop codon *via* MAGE. The pKD-SepRS-EFSep plasmid was maintained using kanamycin resistance and both SepRS and EFSep were induced using IPTG. The pSepT plasmid was maintained using tetracycline resistance, and $tRNA^{Sep}$ was constitutively expressed.

Effect of RF1 deletion, aaRS expression, and NSAA incorporation on fitness: Stationary phase pre-cultures were obtained by overnight growth with shaking at 34 °C in 150 µl LB^L supplemented with chloramphenicol for plasmid maintenance. Stationary phase cultures were diluted 100-fold into 150 µl LB^L containing chloramphenicol and 0.2% L-arabinose and/or 1 mM pAcF where indicated. Growth was monitored on a Biotek Synergy H1 plate reader. OD_{600} was recorded at 10-minute intervals for 16 hours at 34 °C with continuous shaking. All data were measured in triplicate. Doubling time was determined for each replicate as described above, and replicates were averaged for Figure 5-3A.

GFP variant synthesis: GFP variants (Table S5-33) were synthesized as gBlocks by IDT and modified with an N-terminal 6His tag *via* PCR. His-tagged GFP variants were isothermally assembled (*39*) into the pZE21 plasmid backbone (*40*) to yield the array of GFP reporter plasmids used in this study. Reporter plasmids were maintained using kanamycin resistance and induced using 30 ng/mL anhydrotetracycline (aTc).

UAG suppression and GFP Fluorescence: Stationary phase pre-cultures were obtained by overnight growth with shaking at 34 °C in 150 µl LB^L supplemented with appropriate antibiotics for plasmid maintenance. Stationary phase cultures were diluted 100-fold into 150 µl fresh LB^L containing the same antibiotics as the overnight pre-culture. These cultures were grown to mid-log phase and diluted 100-fold into 150 µl fresh LB^L containing the same antibiotics plus 30 ng/ml aTc, 0.2% L-arabinose, and/or 1 mM pAcF (where indicated). Protein expression proceeded for 16 hours at 34 °C with continuous shaking. Following 16 hours of expression, cultures were transferred to V-bottomed plates, pelleted, and washed once in 150 µL of PBS (pH 7.4). Washed pellets were resuspended in 150 µL of PBS (pH 7.4) and transferred to a black-walled, clear-bottom plate to measure GFP fluorescence for each strain. Both OD_{600} and GFP fluorescence (Ex: 485 nm, Em: 528 nm) were measured on a Biotek Synergy H1 plate reader. Fluorescence and OD_{600} measurements were corrected by subtracting background fluorescence and OD_{600} (determined using PBS blanks). Relative fluorescence (in rfu) was calculated by the ratio fluorescence/OD_{600}. Reported values represent an average of four replicates. After

measurements were complete, the cells were pelleted, the supernatant was aspirated, and the pellets were frozen at -80 °C for subsequent protein purification and Western blot analysis.

Protein extraction and Western blots: Cell pellets were obtained as described above. Cells were lysed using a lysis cocktail containing 150 mM NaCl, 50 mM Tris-HCl, 0.5x BugBuster reagent, 5% glycerol, 50 mM Na_3VO_4, 50 mM NaF, protease inhibitors (Roche), and 1 mM DTT. The resulting lysates were spun at 4 °C for 15 minutes at 3200 x g only in cases where soluble and insoluble fractions were separately analyzed. Protein lysate concentrations were determined using the BioRad-DC colormetric protein assay. Lysates were normalized by optical density at 600 nm, resolved by SDS-PAGE, and electro-blotted onto PVDF membranes (Millipore, # ISEQ00010). Western blot analysis was performed with mouse monoclonal antibody directed against GFP (Invitrogen, # 332600), and membranes were imaged with an HRP secondary antibody (Jackson Immunoresearch, JAC-715035150) *via* chemiluminescence on a ChemiDoc system (BioRad).

Mass spectrometry

Materials: Urea, Tris-HCl, $CaCl_2$, iodoacetamide (IAA), Pyrrolidine, DL-lactic acid, HPLC grade water and acetonitrile (ACN) were from Sigma-Aldrich (St. Louis, MO). Chloroform and dithiothretitol (DTT) were from American Bioanalytical (Natick, MA). Methanol, trifluoroacetic acid (TFA), ammonium hydroxide and formic acid (FA) were obtained from Burdick and Jackson (Morristown, NH). Sequencing grade modified trypsin was from Promega (Madison,WI). Anionic acid cleavable surfactant II (ALS) was from Protea (Morgantown, WV). UltraMicroSpin^{TM} columns, both the C_{18} and the DEAE PolyWAX variety were from The Nest Group, Inc. (Southborough, MA). Titaniumdioxide (TiO_2) with a particle size of 5 µm was obtained from GL Sciences Inc. (Torrance, CA).

Cell culture and lysis: Strains were routinely grown in LB^L media with the following concentration of antibiotics when appropriate: tetracycline (12 µg/mL), kanamycin (50 µg/mL), chloramphenicol (12 µg/mL), and zeocin (25 µg/mL). Bacterial cell cultures were grown at 30°C while shaking at 230 rpm until late log phase, quenched on ice and pelleted at 10,000 x g (10 min). The media was discarded and the cell pellets were frozen at -80°C to assist with subsequent protein extraction. Frozen cell pellets were thawed on ice and lysed in lysis buffer consisting of BugBuster reagent, 50 mM Tris-HCl (pH 7.4, 23°C), 500 mM NaCl, 0.5 mM EGTA, 0.5 mM EDTA, 14.3 mM 2-mercaptoethanol, 10 % glycerol, 50 mM NaF, and 1 mM Na_3O_4V, Phosphatase inhibitor cocktail 3 and complete protease inhibitor cocktail (Sigma Aldrich) were added as recommended by the corresponding manufacturer. Cell suspensions were incubated on ice for 30 min and the supernatant was removed after ultracentrifugation. The remaining pellet was re-extracted and resulting fractions were combined.

Protein lysates: Protein was precipitated with the methanol/chloroform method as previously described (*41*). One third of the resulting protein pellet was dissolved in 1.5 ml freshly prepared 8 M Urea/0.4 M Tris-HCl buffer (pH= 8.0, 23 °C). 5 mg protein was reduced and alkylated with IAA and digested overnight at 37°C using sequencing grade trypsin. The protein digest was desalted using C_{18} Sep-Pak (Waters) and the purified peptides were lyophilized and stored at -80°C.

Digestion of intact E. coli for shotgun proteomics: Cells were grown overnight to stationary phase, quenched on ice, and 2 ml culture was used for protein extraction and mass spectrometry. Cells were pelleted for 2 min at 2000 x g and the resulting pellet was washed twice with 1 ml ice cold Tris-HCl buffer pH=7.4, 23°C. The cells were then re-suspended in 100 µl Tris-HCl buffer pH=7.4, 23°C, split into 4 equal aliquots of 25 ul and the cell pellet was frozen at -80 °C. Frozen pellets were lysed with 40 µl lysis buffer consisting of 10 mM Tris-HCl buffer pH = 8.6 (23°C) supplemented with 10 mM DTT, 1 mM EDTA and 0.5 % ALS. Cells were lysed by vortex for 30 s and disulfide bonds were reduced by incubating the reaction for 35 min. at 55 °C in a heating block. The reaction was briefly quenched on ice and 16 µl of a 60 mM IAA solution was added. Alkylation of cysteines proceeded for 30 min in the dark. Excess IAA was quenched with 14 µl of a 25 mM DTT solution and the sample was then diluted with 330 µl of 183 mM Tris-HCl buffer pH=8.0 (23 °C) supplemented with 2 mM $CaCl_2$. Proteins were digested overnight using 12 µg sequencing grade trypsin for each protein aliquot, and the reaction was then quenched with 64 µl of a 20 % TFA solution, resulting in a sample pH<3. Remaining ALS reagent was cleaved for 15 min at room temperature. An aliquot of the sample consisting of ~30 µg protein (as determined by UV_{280} on a nanodrop) was desalted by reverse phase clean-up using C_{18} UltraMicroSpin columns. The desalted peptides were dried at room temperature in a rotary vacuum centrifuge and reconstituted in 30 µl 70 % formic acid 0.1 % TFA (3:8 v/v) for peptide quantitation by UV_{280}. The sample was diluted to a final concentration of 0.6 µg/µl and 4 µl (2.4 µg) were injected for LC-MS/MS analysis of the unfractionated digest using a 200 min method.

Phosphopeptide enrichment: Offline phosphopeptide enrichment was carried out with Titanium dioxide (TiO_2) using a bulk enrichment strategy adapted from Kettenbach *(42)*. Briefly, between 0.4 and 1 mg of desalted peptide digest was transferred into a 1.5 ml PCR tube and dissolved at a concentration of 1mg/ml in "binding solution" consisting of 2 M lactic acid in 50 % ACN. Activated TiO_2 was prepared as a concentrated slurry in binding solution and added to the peptide solution to obtain a TiO_2 to peptide ratio of 4:1 by mass. The mixture was incubated for 2 h at room temperature on an Orbit M60 laboratory shaker operated at 140 rpm. The suspension was centrifuged for 20 s at 600 x g and the supernatant was removed. The TiO_2 beads were washed twice with 50 µl of the binding solution and then 3 times with 100 µl 50 % ACN, 0.1 % TFA. Stepwise elution of phosphopeptides from the beads was carried out using 20 µl of 0.2 M sodium phosphate buffer pH=7.8 followed by 20 µl 5 % ammonium hydroxide and 20 µl 5 % pyrrolidine solution. The pH of the combined extracts was adjusted with 30 µl of ice cold 20 % TFA resulting in a sample pH <3.0. Peptides were desalted on C_{18} UltraMicroSpin columns as described above and the peptide concentration was estimated by UV_{280}.

Offline fractionation of tryptic digests: Offline electrostatic repulsion-hydrophilic interaction chromatography (ERLIC) *(43)* was performed on disposable DEAE PolyWAX UltraMicroSpin columns. Columns were activated as recommended by the manufacturer and then conditioned with 3 x 200 µl washes with 90 % ACN, 0.1 % acetic acid (buffer A). For this purpose, the columns were centrifuged for at 200 x g for 1 min at 4°C. The column was then loaded with 50 µg of a desalted peptide digest prepared in 25 µl buffer A, and the flow-through was collected. Stepwise elution of the peptides was carried out using brief centrifugation steps carried out for 30 s at 200 x g with 50 µl eluent unless noted otherwise. The elution steps consisted of the following volumetric mixtures of buffer A and buffer B (0.1 % formic acid in 30 % ACN): (1) 100:0 (2) 96:4 (3) 90:10 (4) 80:20 (5) 60:40 (6) 100 µl of 20:80 (7) 100 µl of 0:100. Additional

elution steps consisted of: (8) 1 M triethylamine buffer adjusted with formic acid to pH=2.0. (9) 0.2 % ammonia (10) 0.2 % ammonia and finally (11) 100 µl 70 % formic acid. The collected fractions were dried in a vacuum centrifuge and reconstituted in 15 µl solvent consisting of 3:8 by volume of 70 % formic acid and 0.1 % TFA. Fractions were analyzed by LC-MS/MS using a 400 min gradient.

Liquid chromatography and mass spectrometry: Capillary LC-MS was performed on an Orbitrap Velos mass spectrometer (Thermo Fisher Scientific) connected to a nanoAcquity UPLC (Waters, Milford, MA). Liquid chromatography was performed at 35 °C with a vented split setup consisting of a commercially available 180 µm x 20 mm C_{18} nanoAcquity UPLC trap column and a BEH130C18 Waters symmetry 75 µm ID x 250 mm capillary column packed with 5 and 1.7 µm particles respectively. Mobile phase A was 0.1 % formic acid (FA) and mobile phase B was 0.1 % FA in acetonitrile. The injection volume was 4-5 µl depending on the sample concentration. Up to 2.4 µg peptides were injected for each analysis. Peptides were trapped for 3 min in 1 % B with and a flow rate of 5 µl/min. Gradient elution was performed with 90, 200 and 400 min methods with a flow rate of 300 nl/min. Two blank injections were performed between samples to limit potential carryover between the runs. The gradient for the 90 min method was 1-12 % B over 2 min, 12-25 % B over 43 min, 25-50 % B over 20 min, followed by 6 min at 95 % B and column re-equilibration in 1 % B. The gradient for the 200 min was 1-10 % B over 2 min, 10-25 % B over 150 min, and 25-50 % B over 20 min, followed by 7 min at 95 % B and re-column equilibration at 1 % B. The gradient for the 400 min was 1 min in 1 % B, 1-7 % B over 2 min, 7-20 % B over 298 min, and 20-50 % B over 60 min, followed by a 1 min flow ramp to 95 % B. The column was flushed for 9 min using 95 % B and then re-equilibrated for 27 min at 1 % B prior to the next injection. Mass spectrometry was performed with a spray voltage of 1.8 kV and a capillary temperature of 270 °C. A top 10 Higher Collisional Energy Dissociation (HCD) method with one precursor survey scan (300-1750 Da) and up to 10 tandem MS spectra performed with an isolation window of 2 Da and a normalized collision energy of 40 eV. The resolving power (at m/z = 400) of the Orbitrap was 30,000 for the precursor and 7500 for the fragment ion spectra, respectively. Continuous lock mass calibration was enabled using the polycyclodimethylsiloxane peak (m/z = 445.120025) as described *(44)*. Dynamic exclusion criteria were set to fragment precursor ions exceeding 3000 counts with a charge state >1 twice within a 30 s period before excluding them from subsequent analysis for a period of 60 s. The exclusion list size was 500 and early expiration was disabled.

Proteomics data processing: Raw files from the Orbitrap were processed with Mascot Distiller and searched in-house with MASCOT (v. 2.4.0) against the EcoCyc *(45)* protein database release 16.0 for *E. coli* K-12 substr. MG1655 with a custom database and search strategy designed to identify amber suppression (Aerni et al. manuscript in preparation). Forward and decoy database searches were performed with full trypsin specificity allowing up to 3 missed cleavages and using a mass tolerance of ±30 ppm for the precursor and ±0.1 Da for fragment ions, respectively. Cysteines were considered to be completely alkylated with IAA unless samples were processed by a gel-based workflow. In that case Propionamide (C) was considered as a variable modification. Additional variable modifications for all searches were oxidation (M) and deamidation (NQ) for samples processed with urea Carbamyl (K, R, N-term). In order to detect pAcF containing peptides, a variable custom modification for Y was introduced with the composition C_2H_2 and monoisotopic mass of 26.015650 Da. Typical FDR were <1 % for

peptides above identity threshold and <2% considering all peptides above identity or homology threshold respectively. The MASCOT search results were deposited in the Yale Protein Expression Database (YPED) *(46)*. The following filter rules were specified in YPED for reporting of protein identifications: (i) At least 2 bold peptides and peptide scores ≥20 or (ii) 1 bold red peptide with a peptide score ≥20 with at least one additional bold red peptide with a score between 15 and 20.

Bacteriophage assays

For all phage experiments, growth was carried out in LB^L at 30 °C. Liquid cultures were aerated with shaking at 300 rpm. Before each experiment, a fresh phage lysate was prepared. To do this, *Escherichia coli* MG1655 was grown to mid-log phase in 3 mL of LB^L, then ~2 uL of T7 bacteriophage (ATCC strain BAA-1025-B2) or T4 bacteriophage (ATCC strain 11303-B4) was added directly from a glycerol stock into the bacterial culture. Lysis proceeded until it was complete (lysate appears clear after ~4 hours). The entire lysate was centrifuged to remove cell debris (10,000 rcf, 10 minutes), and 3 mL of lysate was transferred to a glass vial supplemented with 150 mg NaCl for phage preservation. Lysates were prepared fresh, titered, and stored at 4 °C for the duration of each experiment. One lysate was used for all replicates of a given experiment.

Phage titering: Phage lysate was titered by serial dilution into LB^L (10-fold dilution series). Before plating on LB^L agar, 10 µL of the diluted phage lysate was mixed with 300 µL of mid-log *E. coli* MG1655 culture and 3 mL of molten top agar. Plaques matured for ~4 hours at 30 °C. Titers (pfu/mL) were calculated based on the lysate dilutions that produced 20-200 pfu.

Plaque area: For plaque area assays, bacterial cultures were grown to mid-log phase in 3 mL LB^L. To accommodate different doubling times, faster-growing cultures were continually diluted until all strains reached OD_{600} ~0.5. Immediately prior to infection, OD_{600} was normalized to 0.50 for all cultures. Approximately 30 pfu of T7 bacteriophage were mixed with 300 µL of OD_{600} = 0.50 culture and 3 mL of molten top agar, and then immediately plated on LB^L agar. Plaques were allowed to mature at 30 °C for 7 hours, then the plates were imaged on a Bio-Rad Gel Doc system, and plaque areas were measured using ImageJ (*47*). Statistics were based on a Kruskal-Wallis one-way ANOVA followed by Dunn's multiple comparison test, where *$p < 0.05$, **$p < 0.01$, and ***$p < 0.001$.

T7 Fitness: Fitness was assessed in triplicate at low MOI based on protocols by Heineman *et al.* (*22*). Briefly, bacterial glycerol stocks were inoculated directly into 3 mL LB^L and serially diluted in LB^L. Serial dilutions were grown overnight (30 °C, 300 rpm), so that one of the dilutions would be at mid-log growth phase in the morning. Prior to infection, a second dilution series was performed so that host strains would be at optimal growth phase over the course of the serial infection. Starting cultures were normalized to OD_{600} = 0.50 by adding LB^L immediately before infecting the cultures (MOI = 0.015) at t = 0. Infected culture was diluted 1/10 into 3 mL of uninfected mid-log phase culture at 30 minute intervals. Aliquots of the infection were taken at t = 4, 10, 60, and 120 minutes. At t = 4, the aliquot was treated with chloroform to quantitate non-adsorbed phage particles. For all other time points (t = 10, 60, and 120), aliquots were immediately mixed with 300 µL of mid-log *E. coli* MG1655 and 3 mL molten top agar and then

spread on LB^L agar. Plaques were counted after maturing for ~4 hours at 30 °C, and then pfu/mL was calculated for each time point, correcting for dilutions. Adsorption efficiency was consistently >95% as determined by $(N_{t=4} - N_{t=10}) / N_{t=10}$, and fitness was determined by $[\log_2(N_{t=120}/N_{t=60})]/(\Delta t/(60 \text{ min/hr}))$, where N is the number of phages at time t minutes and Δt = 60 min.

Kinetic lysis time: Mean lysis time was determined with 12 replicates based on protocols from Heineman *et al.* (*22*), except that OD_{600} was monitored instead of OD_{540}. Mid-log phase cells (as in the fitness assay) were infected at MOI = 5, then 150 µL aliquots of infected culture were distributed into a 96-well flat bottomed plate and sealed with Breathe-Easy™ sealing membrane. Lysis was monitored at 30 °C with shaking at 300 rpm on a Biotek H4 plate reader with OD_{600} measurements taken every 5 minutes. Each lysis curve was fit to a cumulative normal distribution using the normcdf function in MATLAB. Mean lysis time, mean lysis OD_{600}, and mean lysis slope were calculated using this cumulative normal distribution function.

B. Time and cost

In order to demonstrate the efficiency and cost-effectiveness of our recoding strategy, we explicitly present the total full time equivalents (10.75 FTE years) and DNA costs ($20,333) required to complete this project. Because much of our research time was spent developing and optimizing these genome engineering tools as described below, we estimate the actual time spent constructing a fully recoded genome (5.5 FTE years), and the minimum amount of time that it would take to repeat its construction with current knowledge (0.5 FTE years) (Tables S10 and S11). By contrast, the design, synthesis, and assembly of the 1.08–mega–base pair *Mycoplasma mycoides* JCVI-syn1.0 genome required $40 million and more than 200 FTE years (*48*). While future *de novo* genome synthesis projects will likely improve on these figures by incorporating chip-based DNA synthesis (*49*), our strategy nevertheless demonstrates considerable advantages in the cost and efficiency of making hundreds of genome changes.

Table S5-17. Time required to reassign UAG

Phase	Technology development	Actual strain construction	Time to repeat	Time with CoS-MAGE[a]
MAGE	3.75	1.50	0.15	0.24
CAGE	7.00	4.00	0.35	0.12
Total	**10.75**	**5.50**	**0.50**	**0.36**

[a]Suggested improvements: make 40 changes per strain using improved CoS-MAGE strains (*50, 51*)

Table S5-18. DNA cost for reassigning the UAG codon

Oligo type	MAGE oligos	mascPCR primers	Cassette amplification primers	Cassette screening primers	Deletion oligos	Total
Description	320 x 90-mer oligos with 4 PTO bonds	978 oligos (~23 bp)	190 x 72-mer oligos	144 x 25-mer oligos	25 x 90-mer oligos	-
Yield	100 nmole DNA plate	25 nmole DNA plate	25 nmole DNA plate	25 nmole DNA plate	100 nmole DNA plate	-
Price per base*	$0.28 per base, $3.50 per PTO bond	$0.18 per base	$0.18 per base	$0.18 per base	$0.28 per base	-
Total price	$12,544.00	$4,048.92	$2,462.40	$648.00	$630.00	$20,333.32

*IDT standard price

Since we developed MAGE and CAGE at the same time as we were using them to reassign UAG, a considerable portion of our effort was devoted to technology optimization and changing strategies. For instance, since *tolC* negative selection yields scarless conjugal junctions, desired conjugants can be prepared for subsequent conjugations in one step by inserting *kanR*-oriT or *tolC* directly into one of the existing positive markers (*11*). Therefore, 6 modular cassettes targeting *kanR*-oriT or *tolC* to replace *specR* (spectinomycin), *zeoR* (zeocin), or *gentR* (gentamycin) are adequate for all conjugations beginning with the second round. Our initial designs did not take this into account, so we first had to remove one or both positive markers via a two step replacement and deletion procedure using *tolC* or *galK*. However, now that we better understand the homology requirements for precisely assembling genome segments of various sizes (Table S5-20), selectable markers can be placed to permit one-step turnaround between conjugations. Therefore, we report both the FTE time required to complete the construction of C321 and the estimated FTE time required to repeat the project with current knowledge (Table S5-17).

Table S5-20. Positions of markers for CAGE and window sizes for conjugal junctions

Conjugation	Donor oriT position	Recipient PN marker position	lo Positive marker position	hi Positive marker position	oriT/tolC junction[a]	Positive marker junction
Conj1	4019968	none	3921005	4417928	undefined	4142298
Conj2	4497524	none	4417928	4612628	undefined	4444521
Conj3	189613	182395	4612628	374608	7218	4238020
Conj4	480320	474528	36400	629000	5792	4046621
Conj5	781100	788054	608541	903110	6954	4344652
Conj6	1145180	1124600	892756	1255700	20580	4276277
Conj7	1416412	1415470	1255700	1542300	942	4352621
Conj8	MAGE	MAGE	MAGE	MAGE	MAGE	MAGE
Conj11	2438300	2428900	2223738	2627100	9400	4235859
Conj12	2784761	2783150	2627100	2840467	1611	4425854
Conj13	2967175	2968028	2840467	3014000	853	4465688
Conj14	3176034	3184259	3010540	3334920	8225	4314841
Conj15	3544352	none	3331657	4245059	undefined	3725819
Conj16	3816822	none	3735445	4245059	undefined	4129607
Conj17	4417928	4417928	3921005	4612628	0	3947598
Conj18	374608	36400	4612628	629000	338208	3983628
Conj19	892756	903110	608541	1255700	10354	3992062
Conj20	1529620	1542300	1255700	1702450	12680	4192471
Conj21	MAGE	MAGE	MAGE	MAGE	MAGE	MAGE
Conj22	2627100	none	2223738	2840467	undefined	4022492
Conj23	3014000	3010540	2840467	3334920	3460	4144768
Conj24	3734278	none	3332800	3921005	undefined	4051016
Conj25	4610360	4612400	3921005	629000	2040	3292005
Conj26	1255700	none	608541	1702450	undefined	3545312

Table S5-20 (Continued).						
Conj27	2223738	2209114	1710450	2840467	14624	3509204
Conj28	3332800	3346270	2840467	3921005	13470	3558683
Conj29	608541	791470	1702450	2627225	182929	924775
Conj30_Cn2	2848625	2840467	1710450	3921005	8158	2428666
Conj30_Cn7	2840467	2209114	1710450	3921005	631353	2428666
Conj30_5	1719000	1663210	608541	3921005	55790	1326757
Conj31	3864420	3921005	1255700	1719000	56585	463300

[a]Undefined means that there was no selection for the desired crossover position during conjugation

Minimal time required to repeat the construction of C321 with current knowledge

MAGE: 40 days

- 2 days of continuous cycling for 18 cycles
- 16 days to screen 32 MAGE populations (screen 2 populations per day)
- 1 day for 7 additional cycles
- 16 days to screen MAGE populations (screen 2 populations per day)
- 5 days to introduce the remaining UAG alleles and screen for desired clones

CAGE: 90 days

- 1 day to prepare selectable marker cassettes
- 1 day to recombine *specR*, *gentR*, or *zeoR* marker into rEc strains
- 1 day to screen for desired recombinants
- 1 day to recombine marker *tolC* or *kanR*-oriT into recombinants
- 1 day to screen for desired double recombinants
- 85 days for 6 conjugations (minimum of 5 days per conjugation, maximum of 2 conjugations per day)
    - Phase 1: 16 conjugations = 40 days
    - Phase 2: 8 conjugations = 20 days
    - Phase 3: 4 conjugations = 10 days
    - Phase 4: 2 conjugations = 5 days
    - Phase 5: 1 conjugation = 5 days
    - Phase 6: 1 conjugation = 5 days

C. Construction of a recoded genome

Starting from EcNR2 (*Escherichia coli* MG1655 Δ*mutS*::*cat* Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*]), we removed 305/321 UAG codons across 32 "rEc" strains. Each strain had 10 adjacent UAG codons that we converted to UAA using MAGE. None of these strains exhibited impaired fitness. We then used CAGE to hierarchically assemble the recoded segments ("Conj" strains) into a fully recoded strain (summarized in Figure 5-2). We identified and overcame several barriers during genome construction. Below, we describe all deviations from our initial design, which was to (1) create 32 strains each with 10 UAG codons replaced by UAA, (2) hierarchically combine adjacent recoded segments into a strain completely lacking UAG, (3) remove release factor 1 (RF1) so that UAG would not cause translational termination. UAG IDs are based on Table S5-16.

MAGE phase:

UAGs that were not converted (false positives from MASC-PCR analysis): (Table S5-16)
rEc4 retained UAGs 4.9 and 4.10.
rEc5 retained UAGs 5.1 and 5.2.
rEc12 retained UAG 12.9.
rEc14 retained UAG 14.5.
rEc15 retained UAG 15.8.
rEc19 retained UAG 19.7.
rEc30 retained UAG 30.3.

UAGs that were converted in addition to the targeted set (Probably from MAGE oligo mix-ups): (Table S5-16)
rEc29 had UAG→UAA 16.1-16.4, 30.5
rEc30 had UAG→UAA 6.7
rEc31 had UAG→UAA 6.7

CAGE phase:

CAGE design for Conj1, Conj2, Conj3, Conj31, and Conj32: We were still optimizing the conjugation selection criteria at the beginning of the CAGE phase. For the first few conjugations, we used no selections or positive selections at conjugal junctions. As the CAGE phase proceeded, we adopted *tolC* negative selection at the conjugal junction between recoded genome segments to permit scarless genome assembly.

Conj8 MAGE construction: Instead of conjugating rEc15 + rEc16 to produce Conj8, we performed additional MAGE cycling in rEc15 to convert 16.1-16.4. This strain was renamed Conj8, and rEc16 was not used in the final recoded genome assembly.

Conj11 IS insertion into tolC: IS5 was inserted into *tolC* rather than the desired *tolC* deletion. This undesired feature was automatically lost during Conjugation 22.

Conj21 and Conj23 dysfunctional *tolC*: Robust negative selection is important for creating scarless conjugal junctions while ensuring that all donor alleles are transferred during CAGE (*11*). We previously reported that two of our 1/8 recoded strains (Conj21 and Conj23) were found to simultaneously survive positive selection (SDS resistance) and negative selection (Colicin E1 resistance). We were able to correct this phenotype in Conj23 by removing the dysfunctional *tolC* cassette and introducing a functional *tolC* elsewhere in the genome; however, the dysfunctional allele appeared to map elsewhere in Conj21 (*11*). Additionally, the Conj10 parental strain used to create Conj21 appeared to also have the dysfunctional *tolC* phenotype. Since we were unable to readily identify the causative allele via whole genome sequencing (obvious candidate genes such as *tolQ*, *tolR*, *tolA*, and *butB* were not mutated), we re-made Conj21 using CoS-MAGE (*14*). This process took 8 cycles of CoS-MAGE and MASC-PCR screening (25 calendar days) to convert 30 UAG codons to UAA. We have found that using PCR to confirm the loss of *tolC* during conjugation is generally adequate to ensure robust isolation of a desired genotype when it is present at a frequency of greater than 1E-5 in the pre-selection population. Therefore, it is advantageous to perform the post-conjugation positive selections first to remove undesired genotypes from the population prior to Colicin E1 selection. Additionally, we are currently working on identifying dysfunctional *tolC* alleles with the goal of mitigating escape mechanisms and thereby increasing the selective power of the *tolC* negative selection.

Potential recombination hotspot caused UAGs to be retained in Conj6, Conj19, and Conj26: Although rare, several UAG codons were unexpectedly retained (Table S5-16) during CAGE despite proper *tolC*/*kan*^{R}-oriT conjugal junction placement. The Conj6 donor failed to transfer UAGs 10.6 – 10.10 during Conjugation 19. In turn, the Conj19 donor failed to transfer UAGs 9.4, 9.5, 9.10, 10.5, 11.3, and 11.8 during Conjugation 26 (S[UAGs converted in all strains]). This region may be a recombination hotspot that promotes several crossovers. We used MAGE to convert these undesired UAGs.

Conj25 *tolC* positive selection: We introduced *tolC* into the Conj18 donor instead of the Conj17 recipient for Conjugation 25. Therefore, we performed SDS selection rather than ColE1 selection. After isolating a desired Conj25 clone, we removed the *tolC* via λ Red before replacing selectable markers and proceeding to the next conjugation.

Conj20, Conj26, and Conj29 putative rearrangement: We found that *tolC* repeatedly recombined into an unknown location when we attempted to use it to delete *spec*^{R} from Conj20. Therefore, we performed Conjugation 26 without *tolC* negative selection. Unfortunately, the same *tolC* mistargeting was observed in strain Conj26, indicating that the genome feature causing the mistargeting had been inherited. Therefore, we identified the position of the undesired *tolC* insertion so that we could remove it. To this end, we first tested several different selectable cassettes and found that the *tolC* cassette's promoter and terminator sequences were both necessary and sufficient for the Conj20 mistargeted *tolC* insertion (Figure S5-11A).

Next, we used inverse PCR and Sanger sequencing to locate the exact position of the *tolC* mistargeting (Figure S5-11B). Briefly, we purified genomic DNA from a Δ*tolC*:*kan*^{R} recombinant, sheared it to ~2 kb fragments on a Covaris AFA Ultrasonication machine, end repaired the gDNA fragments (NEB end repair kit), and ligated standard Illumina adapters (T4 DNA Ligase). We then used 3 cycles of nested PCR in which one primer annealed to the

Illumina adapter and the other 3 primers annealed facing outwards from the *kan*^{R} gene to amplify each junction between the *kan*^{R} gene and the surrounding genomic sequence. We gel purified the portion of the smear corresponding to ~1 kb PCR products, re-amplified with the third nested primer, purified the product (Qiagen PCR purification kit), and then directly Sanger sequenced without subcloning to identify the genome sequence flanking *tolC*. Sequencing indicated that the *kan*^{R} N-terminus was inserted just downstream of nt 3,176,063 (endogenous position of *tolC*), and that the *kan*^{R} C-terminus was inserted just upstream of nt 3,421,404. These loci are 245,341 kb apart in the *E. coli* MG1655 genome. Although we were unable to identify the structural variant in Conj20 via whole genome sequencing, we confirmed the putative rearrangement via colony PCR using primers that hybridize ~150 bp on either side of the putative *kan*^{R} insertion site, and we observed the expected 1.5 kb amplicon (1.2 kb *kan*^{R} + 300 bp of flanking genome sequence, verified via Sanger sequencing). The same PCR in Conj20 (without Δ*tolC*:*kan*^{R} inserted) did not produce an amplicon, and PCR amplification of the endogenous *tolC* locus of both strains produced the expected amplicon for a *tolC* deletion. Taken together, this indicates that the region near the endogenous *tolC* was duplicated and inserted near nt 3,421,404, that a large sequence (too large to be detected by PCR) is deleted by Δ*tolC*:*kan*^{R}, and that the endogenous *tolC* region was not impacted by the mistargeting.

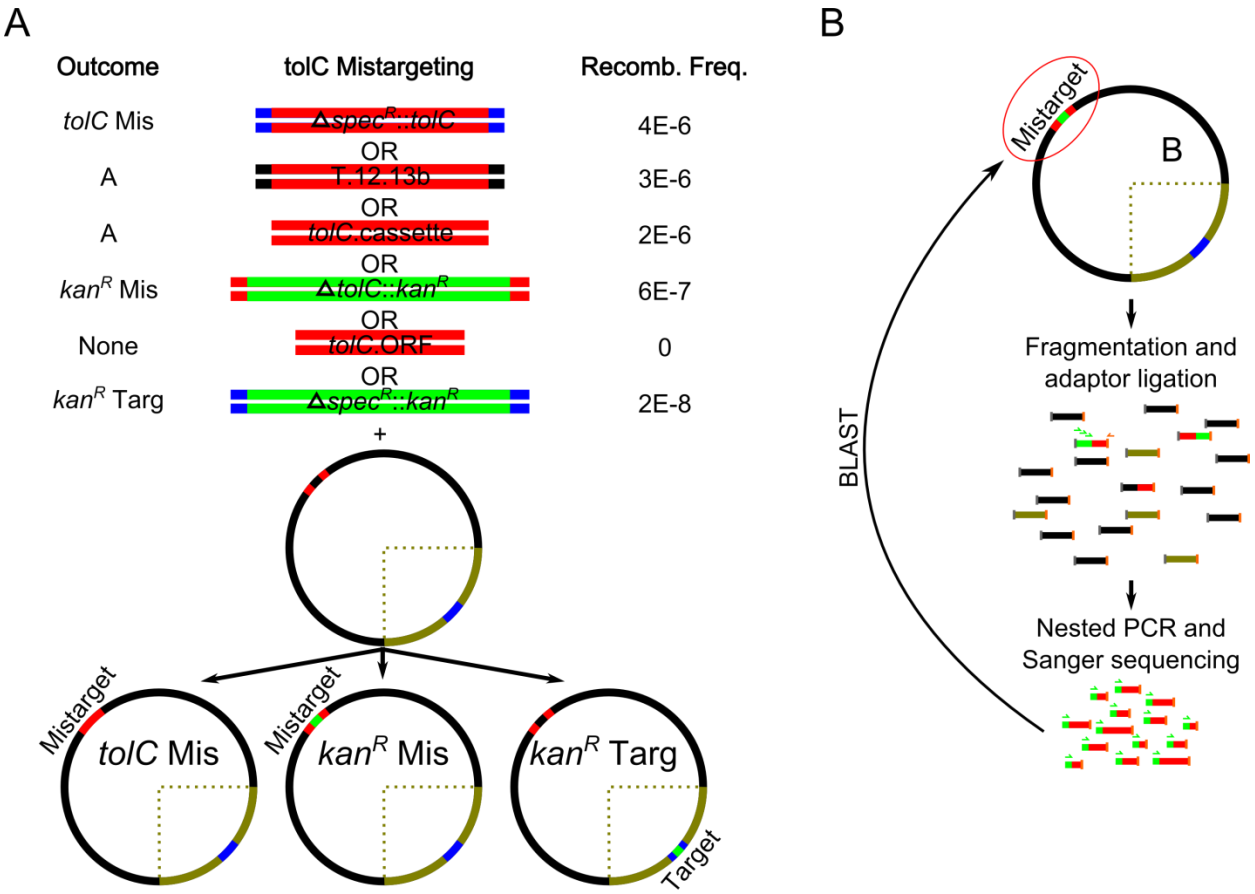


Figure S5-11. Putative Conj20 rearrangement causing *tolC* mistargeting. (A) Several different *tolC* cassettes repeatedly recombined into an unknown locus (*tolC* Mis), a *kan*^{R} cassette having homology to the *tolC* cassette's promoter and terminator sequences efficiently recombined into an unknown locus (*kan*^{R} Mis), a *kan*^{R} cassette having homology to *spec*^{R} efficiently recombined

Figure S5-11 (Continued). into the expected locus (*Kan*^{R} Targ), and the *tolC* ORF lacking a promoter and terminator was not recombinogenic in Conj20. Therefore, the *tolC* cassette's promoter and terminator were necessary and sufficient to mediate *tolC* mistargeting in Conj20. (B) The position of mistargeting was identified by purifying the genome of C20.DT:*kan*^{R}, fragmenting to ~2 kb pieces on a Covaris AFA Ultrasonication machine, repairing DNA ends with a NEB End Repair kit, ligating Illumina adaptors, and performing 3 rounds of nested inverse PCR. The amplicons were gel purified, re-amplified, Sanger sequenced, and BLASTed against the *E. coli* MG1655 genome (taxid:511145). The N-terminal insertion site was nt 3,176,063 (endogenous *tolC* position) and the C-terminal insertion site was nt 3,421,404 (245,341 bp away).

Since the putative rearrangement in Conj20 and Conj26 (region including nt 3,176,063 – nt 3,421,404) was distant from the recoded region (nt 633,969 – nt 1,663,144), we easily prevented its transfer during Conjugation 29 by placing the Conj25 recipient's positive selectable marker at SIR.22.23c (nt 2,627,225) instead of SIR.32.1 (nt 3,921,005). This marker placement permitted a *tolC*/*kan*^{R}-oriT junction between nt 608,541 – nt 629,000 (20,459 bp) and a *gent*^{R}/*zeo*^{R} junction between nt 1,702,450 and nt 2,627,225 (924,775 bp) (Figure S5-12).

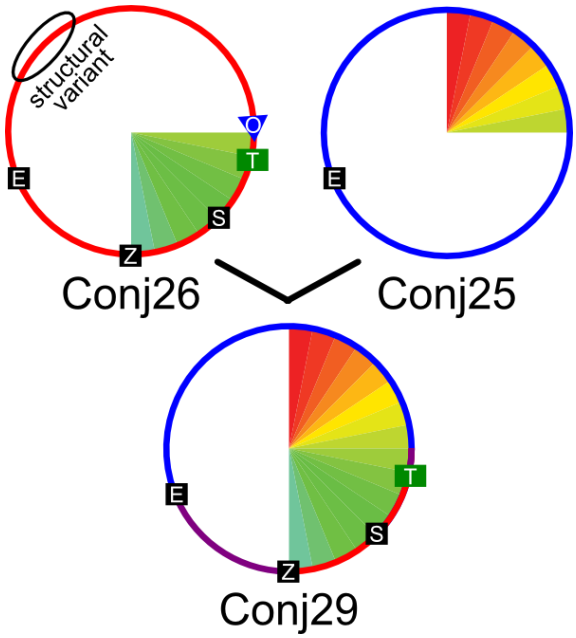


Figure S5-12. Strategic marker placement allowed removal of the undesired structural variant from Conj26. Rather than placing *gent*^{R} at the boundary of the Conj25 recoded region, it was placed further away to select against inheritance of the Conj26 structural variant. Red lines represent Conj26 donor genome sequence, blue lines represent Conj25 recipient genome sequence, and purple lines indicate conjugal junction regions.

Inadequate homology for conjugal junction in Conj28 and Conj30: There is an average of 14.3 kb spanning adjacent UAG codons in *E. coli* MG1655, but many of these regions are inadequate for transferring large genome segments, since conjugal transfer frequency decreases exponentially with increasing distance (*52*). Our first attempts at using small homology regions to transfer large genome segments either led to failed selections (Conj28) or produced low

complexity populations consisting of few recombinants (Conj30). By increasing the distance between *kan*^{R}-oriT and *tolC* (Table S5-20), complete transfer of the recoded segment was achieved, but marker placement sometimes allowed recoded alleles near conjugal junctions to be lost (Figure 5-2, Table S5-16).

Our initial attempts at Conjugation 28 failed because 2120 bp of homology between the donor's *kan*^{R}-oriT and the recipient's *tolC* were inadequate to transfer all 573,882 bp of recoded donor DNA. Instead, the putative Conj28 candidates all retained *tolC* and 25 or more UAG codons proximal to *kan*^{R}-oriT. Therefore, our selections yielded the dysfunctional *tolC* phenotype that was described above for Conj21 and Conj23. However, when we moved *tolC* so that it was 13,470 bp away from *kan*^{R}-oriT and repeated Conjugation 28, we easily selected desired clones.

In another case, the inefficient 1/4 genome transfer during Conjugation 30 yielded a low complexity population retaining 30 undesired UAGs (segments 18-20) in the middle of the donor region (Figure S5-13). Such double crossovers may be caused by two separate conjugations (*52*), or may be formed when the excised recipient genome is partially degraded and recombined back into the donor segment that originally displaced it (*53*). Although the selections did not fail, recombination occurred rarely in the desired 8,158 bp *tolC*/*kan*^{R}-oriT conjugal junction, yielding a single isogenic population (46 out of 46 screened clones) retaining the same 30 UAGs from segments18-20. Rather than repeating the conjugation with the original conjugants, we chose a clone from the first conjugation to carry forward as the recipient in a second conjugation. We moved the selectable markers in Conj27 and the new recipient so that there would be 631,353 bp between *tolC* and *kan*^{R}-oriT, and then repeated the conjugation. This time, all remaining alleles were properly transferred (Figure S5-13).

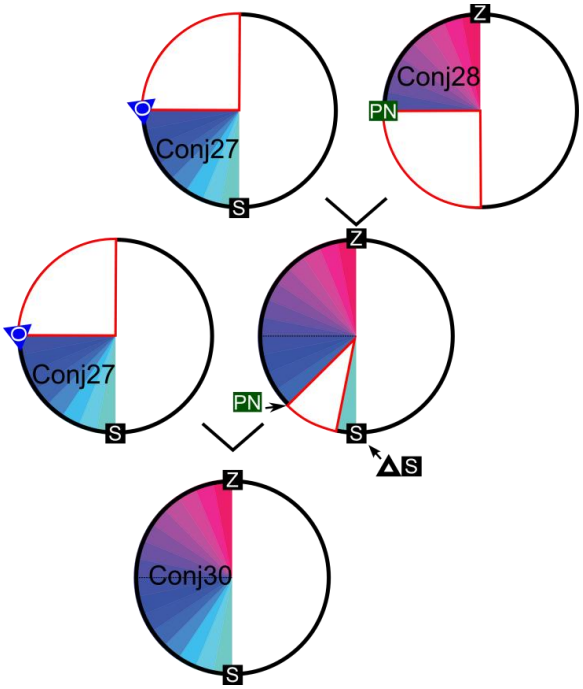


Figure S5-13. Strain Conj30 was prepared by two serial conjugations. The first Conjugation 30 was performed using Conj27 and Conj28 (with 8,158 bp of homology between *tolC* and *kan*^{R}-oriT). After selecting for *Spec*^{R}, *Zeo*^{R}, and ColE1^R, 46 out of 46 clones retained ~30 UAG codons

Figure S5-13 (Continued). in sets 18-20. After removing *spec*^{R} (replaced with *tolC* and then deleted *tolC*) and inserting a new *tolC* near the remaining UAG alleles in the conjugal progeny (providing 631,353 bp between *tolC* and *kan*^{R}-oriT for proper recombination), we performed a second conjugation to transfer the remaining alleles to produce Conj30.

Redundant recoding for Conjugation 31: Based on the above results, the 16.2 kb (*kan*^{R}-oriT/*tolC*) and 61.5 kb (*gent*^{R}/*spec*^{R}) conjugal junctions originally planned for Conjugation 31 were unlikely to accommodate transfer of 1/2 of the genome. Therefore, prior to attempting Conjugation 31, we transferred 1029 kb of recoded genome from Conj26 into Conj30 (C30.5, Figure 5-2) so that this region would be redundantly recoded in both parental strains for Conj31. Additionally, to decrease the chance of a failed *tolC* selection, we inserted *tolC* into the donor strain so that we could positively select on SDS. Thus, Conjugation 31 was successfully performed using a 56.6 kb oriT/*tolC* junction and a 463 kb *gentR*/*specR* junction (Figure 5-2, Figure S5-14).

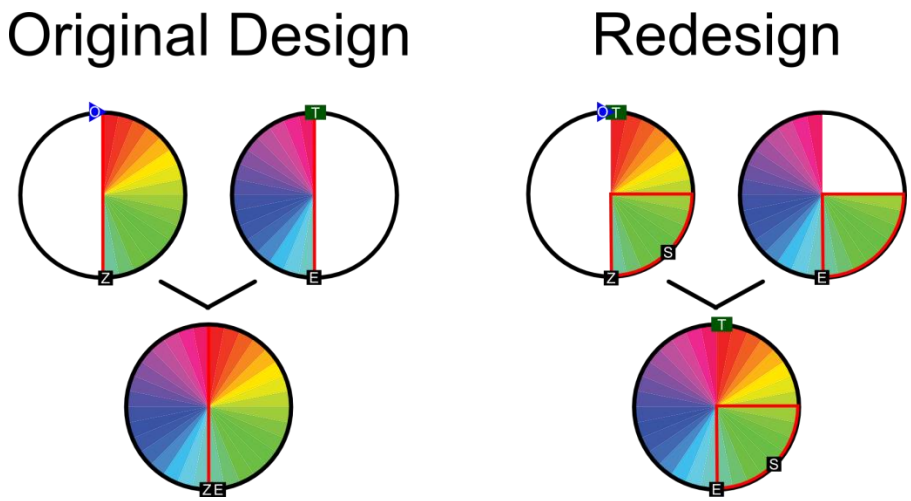


Figure S5-14. Redundant recoding for Conjugation 31. Conj29 and Conj30 only provide 16.2 kb and 61.5 kb of homology for their *kan*^{R}-oriT/*tolC* and *gent*^{R}/*spec*^{R} junctions, respectively. Therefore, we moved the *kan*^{R}-oriT/*tolC* junction and created Conj30.5, which has the third quadrant of the genome redundantly recoded. This provides a 56.6 kb oriT/*tolC* junction and a 463 kb *gentR*/*specR* junction. Additionally, we used *tolC* in the donor genome to permit SDS selection, which has a lower escape rate than ColE1 selection. Colored wedges represent recoded segments containing 10 UAG→UAA conversions, O = *kan*^{R}-oriT, T = *tolC*, E = *gent*^{R}, S = *spec*^{R}.

Removing remaining UAG codons: After the final conjugation, 3 selectable markers (*tolC*, *gent*^{R}, and *spec*^{R}) and 11 UAG codons (Table S5-21) from the original design of 314 UAGs were retained. We used *tolC* to delete these undesired selectable markers and MAGE to convert the UAG codons to UAA.

Table S5-21. UAG codons that were retained in Conj31 after CAGE

Gene	UAG Pos	UAG ID	Trans Dir	Replichore	Why UAG
b4273	4497523	3.10	+	1	Lost during Conjugation 2
ybaA	476249	8.1	+	1	Lost during Conjugation 4
sucB	761962	9.10	+	1	Lost during Conjugation 26
ybiR	853988	10.6	+	1	Lost during Conjugation 19
yceF	1145234	11.10	-	1	Lost during Conjugation 6
ydfP	1637054	15.9	-	2	Lost during Conjugation 20
rzpQ	1647065	15.10	+	2	Lost during Conjugation 20
yegW	2180057	20.8	-	2	Reverted by yegV oligo
ascB	2840436	24.10	+	2	Reverted by Z.24.25 recombination
hycI	2840595	25.1	-	2	Lost during Conjugation 30
atpE	3918973	32.10	-	2	Lost during Conjugation 16

Upon closer inspection, we observed that *yegV* and *yegW* had overlapping, convergent open reading frames so that MAGE oligos individually converting the UAG of one gene would revert the UAG of the other gene. Therefore, we designed a MAGE oligo that would simultaneously convert the UAGs of both *yegV* and *yegW* (Figure S5-15). Such design clashes will become more common as genome designs incorporate more mutations in closer proximity.

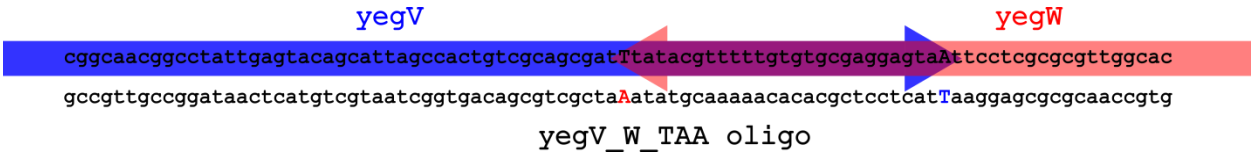


Figure S5-15. MAGE oligo simultaneously converting UAGs of convergently overlapping yegV and yegW genes. The top sequence is the desired genomic sequence (shown 5' → 3'). The bottom sequence is the MAGE oligo that simultaneously converts the UAG codons in *yegV* and *yegW* (shown 3' → 5').

Removing new UAG codons: Genome annotations and interpretations are incomplete and are continually being updated based on empirical results. We initially designed the MAGE oligos based on 314 predicted UAGs (NCBI, NC_000913, Feb. 07, 2006). However, we subsequently identified 8 additional UAGs from the Apr. 24, 2007 NCBI update. Further analysis of the ecocyc.org (*45*) database (Mar. 19, 2012) identified 3 more UAGs (Table S5-22). Ecocyc also flagged 4 previously identified putative UAGs as part of phantom genes (sequences previously annotated, but that are not genes, Table S5-23). We efficiently converted the remaining 11 UAGs via MAGE. However, the fact that we needed to update our design highlights a central problem with using incomplete data to design genomes. Such trivial design changes distributed throughout the genome would require significant effort to implement via whole genome synthesis.

Table S5-22. UAG codons that were not targeted in the original design

Gene	UAG Pos	Trans Dir	Replichore	Identified
yafF	239378	+	1	NC_000913 (NCBI) 02/07/2006 update
yliI	879080	+	1	NC_000913 (NCBI) 02/07/2006 update
ymdF	1067477	+	1	NC_000913 (NCBI) 04/24/2007 update
yheV	3476614	-	2	NC_000913 (NCBI) 04/24/2007 update
yjbS	4266832	-	1	NC_000913 (NCBI) 04/24/2007 update
yjdO	4351104	+	1	NC_000913 (NCBI) 04/24/2007 update
insB	4517037	+	1	NC_000913 (NCBI) 04/24/2007 update
ytjA	4610312	+	1	NC_000913 (NCBI) 04/24/2007 update
mntS	852092	-	1	Ecocyc.org flat file 03/19/2012
yahH	339313	+	1	Ecocyc.org flat file 03/19/2012
ykgN	279248	-	1	Ecocyc.org flat file 03/19/2012

Table S5-23. UAG codons found in genes re-annotated as phantom

Gene	UAG Pos	UAG ID	Trans Dir	Replichore
b4250	4481621	3.8	+	1
b1354	1426575	14.2	+	1
b1367	1433519	14.4	+	1
b2191	2296256	21.5	+	2

Cleanly removing RF1 without impairing fitness: The complete deletion of *prfA* also removes the ribosomal binding site (RBS) from the overlapping essential gene, *prmC*. Therefore, we tested three *prfA* deletion cassettes (Δ*prfA*::*spec*^{R}, Δ*prfA*::*tolC*, and a clean deletion) to remove the ability of UAG to terminate translation. While *spec*^{R} contains an appropriately placed RBS, the C-terminus of *tolC* is C/T rich, so we added a synthetic RBS to ensure adequate *prmC* expression. Finally, we cleanly deleted Δ*prfA:tolC* while retaining the synthetic RBS for *prmC*. All three designs produced viable Δ*prfA* strains without significantly impairing fitness (Figure 5-3).

>Δ*prfA*::*spec*^{R}

ctggagtaacagtacatcattttctttttttacagggtgcatttacgcctatgcgctcacgcaactggtccagaaccttgaccgaacgcagcggt
ggtaacggcgcagtggcggttttcatggcttgttatgactgttttttttggggtacagtctatgcctcgggcatccaagcagcaagcgcgttacg
ccgtgggtcgatgtttgatgttatggagcagcaacgatgttacgcagcagggcagtcgccctaaaacaaagttaaacatcatgagggaagc
ggtgatcgccgaagtatcgactcaactatcagaggtagttggcgtcatcgagcgccatctcgaaccgacgttgctggccgtacatttgtacg
gctccgcagtggatggcggcctgaagccacacagtgatattgatttgctggttacggtgaccgtaaggcttgatgaaacaacgcggcgagc
tttgatcaacgacctttttggaaacttcggcttcccctggagagagcgagattctccgcgctgtagaagtcaccattgttgtgcacgacgacatc
attccgtggcgttatccagctaagcgcgaactgcaatttggagaatggcagcgcaatgacattcttgcaggtatcttcgagccagccacgatc
gacattgatctggctatcttgctgacaaaagcaagagaacatagcgttgccttggtaggtccagcggcggaggaactctttgatccggttcct
gaacaggatctatttgaggcgctaaatgaaaccttaacgctatggaactcgccgcccgactgggctggcgatgagcgaaatgtagtgcttac
gttgtcccgcatttggtacagcgcagtaaccggcaaaatcgcgccgaaggatgtcgctgccgactgggcaatggagcgcctgccggccc
agtatcagcccgtcatacttgaagctagacaggcttatcttggacaagaagaagatcgcttggcctcgcgcgcagatcagttggaagaattt

gtccactacgtgaaaggcgagatcaccaaggtagtcggcaaataatggaatatcaacactggttacgtgaagcaataagccaacttcaggc
gagc

>Δ*prfA*::*tolC*
ctggagtaacagtacatcattttctttttttacagggtgcatttacgcctatgaagaaattgctccccattcttatcggcctgagcctttctgggttc
agttcgttgagccaggccgagaacctgatgcaagtttatcagcaagcacgccttagtaacccggaattgcgtaagtctgccgccgatcgtga
tgctgcctttgaaaaaattaatgaagcgcgcagtccattactgccacagctaggtttaggtgcagattacacctatagcaacggctaccgcga
cgcgaacggcatcaactctaacgcgaccagtgcgtccttgcagttaactcaatccatttttgatatgtcgaaatggcgtgcgttaacgctgcag
gaaaaagcagcagggattcaggacgtcacgtatcagaccgatcagcaaaccttgatcctcaacaccgcgaccgcttatttcaacgtgttgaa
tgctattgacgttctttcctatacacaggcacaaaaagaagcgatctaccgtcaattagatcaaaccacccaacgttttaacgtgggcctggta
gcgatcaccgacgtgcagaacgcccgcgcacagtacgataccgtgctggcgaacgaagtgaccgcacgtaataaccttgataacgcggt
agagcagctgcgccagatcaccggtaactactatccggaactggctgcgctgaatgtcgaaaactttaaaaccgacaaaccacagccggtt
aacgcgctgctgaaagaagccgaaaaacgcaacctgtcgctgttacaggcacgcttgagccaggacctggcgcgcgagcaaattcgcca
ggcgcaggatggtcacttaccgactctggatttaacggcttctaccgggatttctgacacctcttatagcggttcgaaaacccgtggtgccgct
ggtacccagtatgacgatagcaatatgggccagaacaaagttggcctgagcttctcgctgccgatttatcagggcggaatggttaactcgca
ggtgaaacaggcacagtacaactttgtcggtgccagcgagcaactggaaagtgcccatcgtagcgtcgtgcagaccgtgcgttcctccttc
aacaacattaatgcatctatcagtagcattaacgcctacaaacaagccgtagtttccgctcaaagctcattagacgcgatggaagcgggctac
tcggtcggtacgcgtaccattgttgatgtgttggatgcgaccaccacgttgtacaacgccaagcaagagctggcgaatgcgcgttataacta
cctgattaatcagctgaatattaagtcagctctgggtacgttgaacgagcaggatctgctggcactgaacaatgcgctgagcaaaccggtttc
cactaatccggaaaacgttgcaccgcaaacgccggaacagaatgctattgctgatggttatgcgcctgatagcccggcaccagtcgttcag
caaacatccgcacgcactaccaccagtaacggtcataaccctttccgtaactaagaggaataatggaatatcaacactggttacgtgaagca
ataagccaac

>Clean deletion
gggctggagtaacagtacatcattttctttttttacagggtggaggaggaataatggaatatcaacactggttacgtgaagcaataagcc

D. GRO nomenclature and applications

Although for clarity we have assigned informal names to describe our key recoded strains, we have also developed the following GRO nomenclature: C(F/E,M,A)_I, where C is the number of codons instances changed, F/E is the number of codons completely removed from the full genome (F), or all essential genes (E), M is the number of previously essential codon functions manipulated (*e.g.* release factors, tRNAs, aminoacyl-tRNA synthetases), A is the number of codons reassigned to a new amino acid (A_{wt} is wild type function and A_o is without any assigned function), and I is a descriptive index to differentiate strain variants. For example, C7(E1,M1,A_1)_Δ*prfA*::$spec^R$.Δ*mutS*::*tolC* has UAG changed to UAA in all 7 essential genes, has RF1 replaced by $spec^R$, incorporates one NSAA at UAG codons, and has *mutS* replaced by *tolC*. Similarly, C321(F1,M1,A_o)_Δ*prfA*.Δ*mutS*::zeo^R.Δ*tolC* has all 321 known UAG codons changed to UAA, has RF1 cleanly deleted, stalls translation at UAG codons, has *mutS* replaced by zeo^R, and has *tolC* deleted.

Table S5-37. Recoded strains and their genotypes

Strain[a]	GRO nomenclature	Essential codons changed[b]	Total codons changed[c]	Previously essential codon functions manipulated[d]	Expected (obs.) UAG translation function[e]
EcNR2	-	0/7	0/321	None	Stop
C0.B*	C0(M1,A_{wt})_Δ*mutS*::zeo^R.*prfB*	0/7	0/321	$prfB^‡$	Stop
C0.B*.ΔA::S	C0(M2,A_o)_B*.Δ*prfA*::$spec^R$.Δ*mutS*::zeo^R.*prfB*	0/7	0/321	$prfB^‡$, Δ*prfA*::$spec^R$	None (stop*)
C7	C7(E1,A_{wt})_Δ*mutS*::*tolC*	7/7	7/321	None	Stop
C7.ΔA::S	C7(E1,M1,A_o)_Δ*prfA*::$spec^R$.Δ*mutS*::*tolC*	7/7	7/321	Δ*prfA*::$spec^R$	None (sup)
C13	C13(E1,A_{wt})_Δ*mutS*::*tolC*	7/7	13/321	None	Stop
C13.ΔA::S	C13(E1,M1,A_o)_Δ*prfA*::$spec^R$.Δ*mutS*::*tolC*	7/7	13/321	Δ*prfA*::$spec^R$	None (sup)
C321	C321(F1,A_{wt})_Δ*mutS*::zeo^R.Δ*tolC*	7/7	321/321	None	Stop
C321.ΔA::S	C321(F1,M1,A_o)_Δ*prfA*::$spec^R$.Δ*mutS*::zeo^R.Δ*tolC*	7/7	321/321	Δ*prfA*::$spec^R$	None (nc)
C321.ΔA::T	C321(F1,M1,A_o)_Δ*prfA*::*tolC*.Δ*mutS*::zeo^R	7/7	321/321	Δ*prfA*::*tolC*	None (nc)
C321.ΔA	C321(F1,M1,A_o)_Δ*prfA*.Δ*mutS*::zeo^R.Δ*tolC*	7/7	321/321	Δ*prfA*	None (nc)

[a]All strains are based on EcNR2 (*Escherichia coli* MG1655 Δ*mutS*::*cat* Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*]) which is mismatch repair deficient (Δ*mutS*) to achieve high frequency allelic replacement; C0 and C321 strains are Δ*mutS*::zeo^R; C7 and C13 strains are Δ*mutS*::*tolC*; C7, C13, and C321 strains have the endogenous *tolC* deleted, making it available for use as a selectable marker. Spectinomycin resistance (S) or tolC (T) were used to delete *prfA* (A). Bacterial genetic nomenclature describing these strains includes :: (insertion) and Δ (deletion).

[b]Out of a total of 7

[c]Out of a total of 321

[d]*prfA* encodes RF1, terminating UAG and UAA; *prfB* encodes RF2, terminating UGA and UAA; $prfB^‡$ = RF2 variant (T246A, A293E, and removed frameshift) exhibiting enhanced UAA termination (*16*) and weak UAG termination (*17*).

[e]Observed translation function: Stop = expected UAG termination; stop* = weak UAG termination from RF2 variant; sup = strong selection for UAG suppressor mutations; nc = near-cognate suppression in the absence of all other UAG translation function.

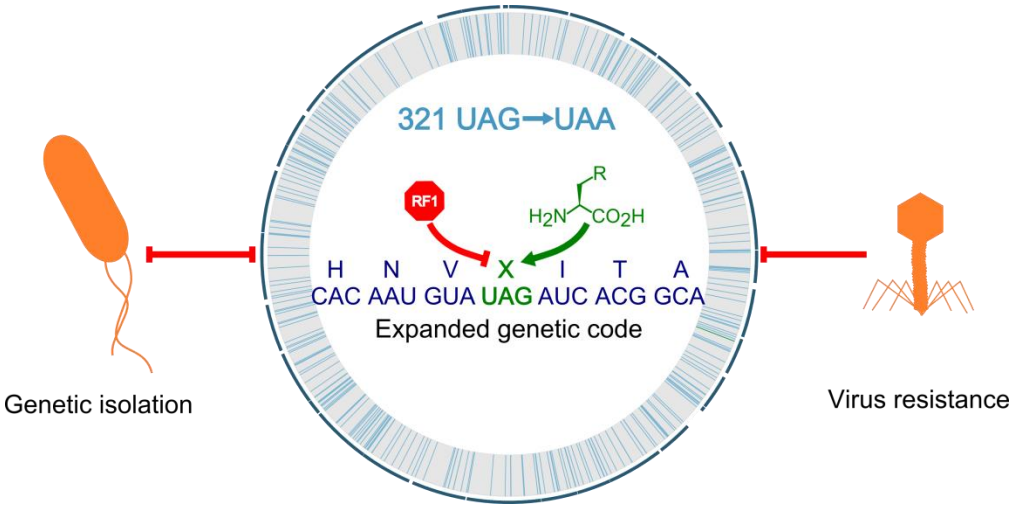


Figure S5-1. Properties of genomically recoded organisms (GROs). We have removed all 321 UAG codons (blue radial lines) and release factor 1 (RF1; terminates translation at UAG) from *E. coli* MG1655. Our recoded strain provides a dedicated UAG codon for plug-and-play translation of nonstandard amino acids (NSAAs). This enables efficient expression of GFP variants containing several UAG codons, provides increased resistance to bacteriophage T7 infection, and establishes a basis for the genetic isolation of GROs.

E. Partial recoding strategies for reassigning UAG codon function

Three hypotheses have attempted to explain why RF1-mediated UAG termination is essential: (i) inadequate RF2-mediated UAA termination (*16, 54*), (ii) essential gene (Table S5-5) loss of function due to UAG read-through (*15*), and/or (iii) translational stalling in the absence of UAG function (*15*). The UAG codon appears to tolerate sense suppression at the majority of UAG codons (*15, 16, 54*). As reported by Mukai et al. (*15*) and illustrated in Figure S5-16, this appears to be an evolutionary feature, given that UAA and UGA stop codons are overrepresented at short distances triplets downstream of UAG codons. We analyzed GO terms using <http://biit.cs.ut.ee/gprofiler/index.cgi>, but we observed no enrichment for any specific component, process, or function.

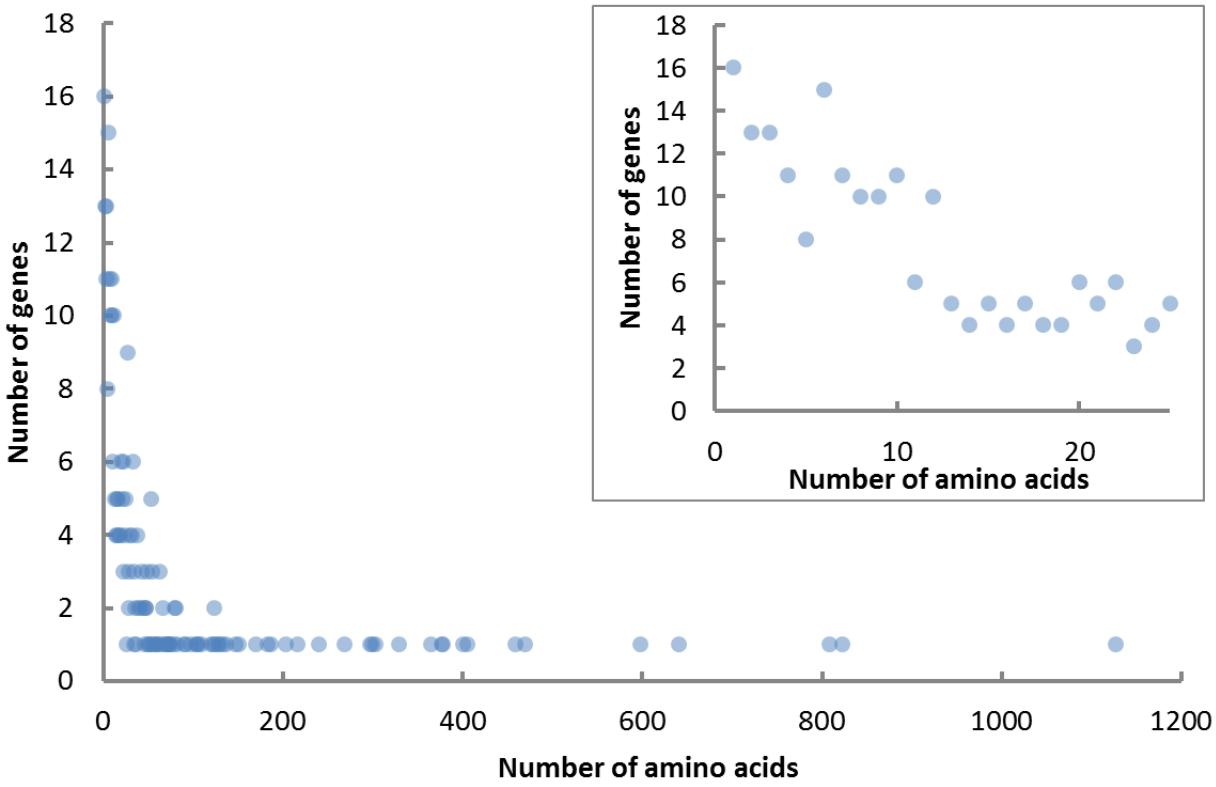


Figure S5-16. Distribution of the number of amino acids added to the C-terminus of genes as a result of UAG read-through. The inset is zoomed in on the first 20 triplets following the UAG codon.

Table S5-5. Essential and important genes terminating with UAG.

Gene	Strand	Gene size (bp)	MG1655 UAG coordinate	Essential[a]	Function[b]	Deletion phenotype[c]
murF	+	1358	96008	Yes	Peptidoglycan biosynthesis	Essential
lolA	+	611	937206	Yes	Periplasmic lipoprotein chaperone	Essential
lpxK	+	986	968575	Yes	LPS biosynthesis	Essential
hemA	+	1256	1264193	Yes	Porphyrin biosynthesis	Essential
hda	-	746	2616097	Yes	Replication initiation regulation	Essential
mreC	-	1103	3396897	Yes	Peptidoglycan biosynthesis and chromosome segregation	Essential
coaD	+	479	3808327	Yes	Coenzyme A biosynthesis	Essential
yafF	+	188	239378	No	Conserved protein, pseudogene	Barely affected
pgpA	+	518	436331	No	Phospholipid processing	Moderate fitness decrease
sucB	+	1217	761962	No	Energy regeneration	Major fitness decrease
fabH	+	953	1148935	No	Fatty acid biosynthesis	Major fitness decrease
fliN	+	413	2019525	No	Component of flagellar motor's switch complex	Moderate fitness decrease
atpE	-	239	3918973	No	Energy regeneration	Major fitness decrease

[a] Essentiality was from the PEC database <http://www.shigen.nig.ac.jp/ecoli/pec/index.jsp> (*55*). Genes in white are essential genes with their UAG replaced in C7.ΔA::S. Genes in gray are additional genes with their UAG replaced in C13.ΔA::S.

[b] Gene functions were referenced from <http://www.ecocyc.org> (*45*).

[c] The deletion phenotype was based on results from the Keio collection (*56*).

F. Analysis of MAGE and CAGE

Doublings

Our recoded strain construction was performed in an EcNR2 background (*Escherichia coli* MG1655 Δ*mutS*::*cat* Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*]), which is defective for mismatch repair. While this background permits efficient allele replacement, it also increases the transition mutation rate ~100 fold. Therefore, continued culturing introduces additional diversity due to spontaneous mutagenesis, which can provide beneficial mutations that compensate for unforeseen genome design flaws. Additionally, these mutations can introduce deleterious mutations that introduce auxotrophies and slow growth, especially when diverse populations are forced through monoclonal bottlenecks (*57*). Although, we have confirmed that the C321.ΔA strains are not auxotrophic, off-target mutagenesis probably underlies their reduced fitness. Therefore, we have calculated the approximate number of doublings for each genome manipulation used in the construction of construction C321.ΔA. Using this information, we estimate the maximum number of doublings during strain construction, the maximum number of doublings that would be expected if we repeated our strain construction, and the maximum number of doublings that would be expected if we improved our strategy by using CoS-MAGE (*14*) to replace 40 UAGs per strain before commencing CAGE. After an estimated 7340 doublings, the 305 off-target mutations detected in C321.ΔA suggests net mutation rate of 9E-9 mutations/bp/doubling, which is consistent with a *mutS*$^{-}$ phenotype (*58*).

MAGE

Step	**Divisions per**	**Number**	**Cell divisions**
MAGE cycles	6	25	150
o/n growths	15	6	90
Re-dilution	5	6	30
Colony/plating	30	2	60
Outgrowth	12	3	36
Dilute, re-grow, freeze	6	1	6
MAGE total			**372**

Selectable marker dsDNA recombinations

Step	**Cell divisions per repetition**
o/n growths	10
Outgrowth, mid-log	10
Induce @ 42, 15 min	0
Electroporation	0
Recover 1 hour	1
Colony/plating	30
colony outgrowth, mid-log	10
Dilute, re-grow, freeze	6
Divisions/Recombination	67
Total	134

Oligo-mediated tolC deletion

Step	Cell divisions per repetition
o/n growths	10
Outgrowth, mid-log	10
Induce @ 42, 15 min	0
Electroporation	0
Recover to stationary	10
Dilute 1/100, outgrowth, mid-log	6
Dilute 1/100, colE1 selection	16[a]
Colony/plating	30
colony outgrowth, mid-log	10
Dilute, re-grow, freeze	6
Divisions/Recombination	98
Total	196

[a]Assumes 1E-3 frequency of tolC deletion

Final MAGE (Conj31->C321.ΔA

Step	Divisions per	Number	Cell divisions
MAGE cycles	6	39	234
o/n growths	15	4	60
Re-dilution	5	14	70
Colony/plating	30	7	210
Outgrowth	12	7	84
Dilute, re-grow, freeze	6	2	12
MAGE total			**670**

CoS-MAGE (off/on cycle):

Step	Cell divisions per repetition
o/n growths	10
Dilute 1/100, outgrowth, mid-log	6
Induce @ 42, 15 min	0
Electroporation (inactivate tolC)	0
Recover for 7 hours, stationary	12
Dilute 1/100, re-growth	6
Dilute 1/100, colE1 selection	14[a]
Colony/plating	30
Outgrowth, mig-log	10
Induce @ 42, 15 min	0
Electroporation (revert tolC)	0
Recover for 3 hours, mid-log	10
Colony/plating	30
Outgrowth	12
Dilute, re-grow, freeze	6
Total	146

[a]Assumes 1% frequency of desired *tolC* genotype

Table S5-2. Total estimated number of doublings required to reassign UAG

	Actual[a]		Re-do[b]		CoS-MAGE[c]	
Manipulation	Number	Doublings	Number	Doublings	Number	Doublings
MAGE	n/a	372	n/a	372	n/a	0
CoS-MAGE	0	0	0	0	3	438
dsDNA Recombinations	19	2546	9	1206	8	1072
Conjugations	7	1792	6	1536	3	768
tolC deletions	10	1960	2	392	3	588
Post-assembly MAGE	n/a	670	n/a	0	n/a	0
Total		7340		3506		2866

[a]Estimated maximum number of actual doublings
[b]Estimated maximum number of doublings to repeat C321.ΔA
[c]Estimated maximum number of doublings using CoS-MAGE to convert 40 UAG codons per strain prior to CAGE

G. Analysis of recoded lineage

Cell morphology in the presence or absence of RF1

Given the extreme degree of genome manipulation necessary to remove all native UAG codons, we wanted to confirm that the cell morphology was not changed (*e.g.* cell elongation or a filamentous phenotype, which might indicate stress response or problems with cell division (*59*). We imaged MG1655, EcNR2, C321, and C321.ΔA::S on bright field using a Zeiss Axio Observer Z1 with a 100X oil immersion objective supplemented with a 1.6X internal lens. Cell morphology was consistent across all strains. The slightly shorter cell lengths for C321 and C321.ΔA::S may be because these strains grow more slowly than MG1655 and EcNR2.

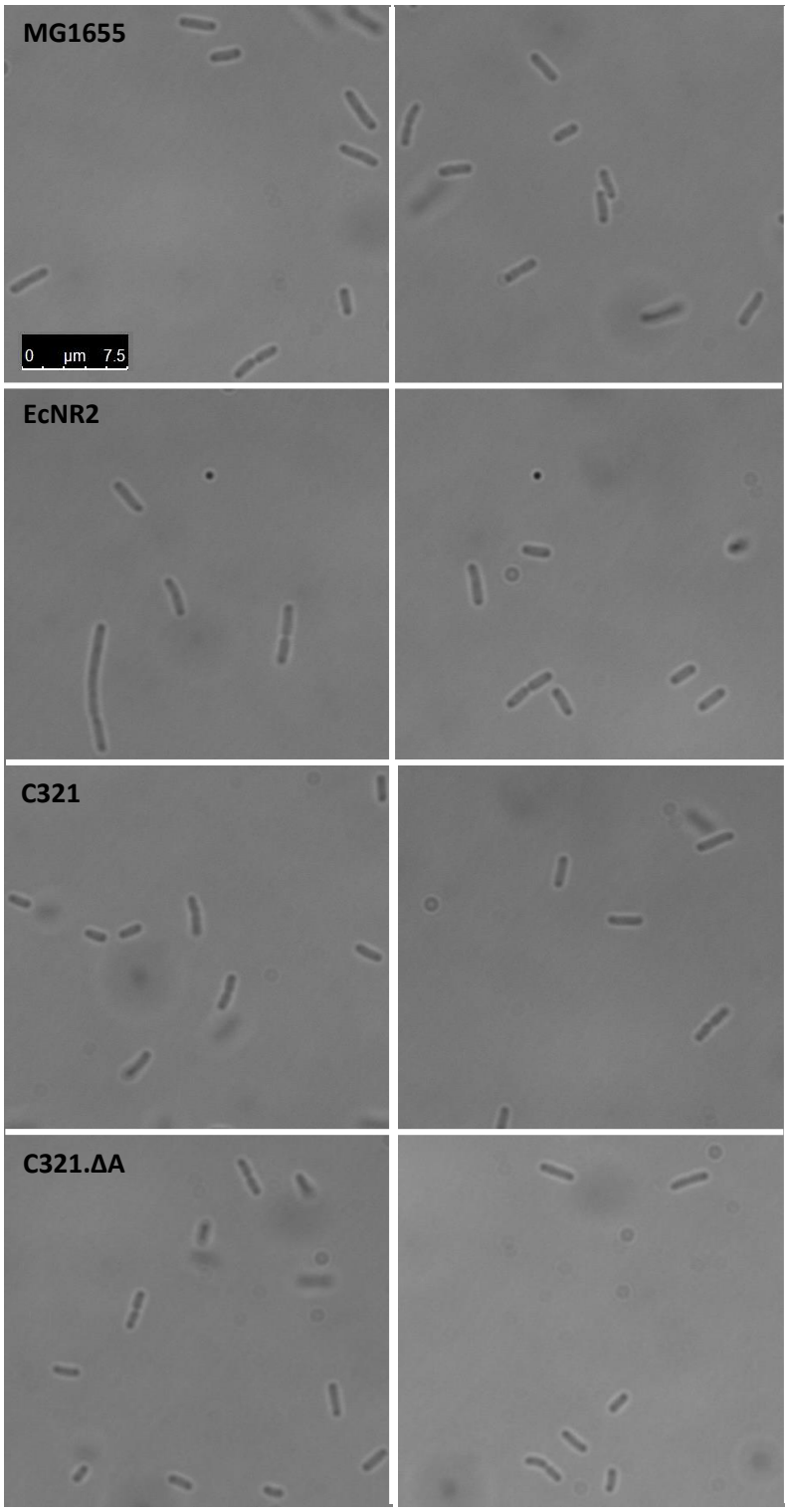


Figure S5-2. Fully recoded strain cell morphology in the presence or absence of RF1. Recoding and RF1 removal does not cause cell aggregation or a filamentous phenotype, which are indictors of cell stress.

Doubling times for each strain in recoded lineage

Doubling times were determined for each strain in the C321.ΔA lineage, represented with a heat map in Figure 5-2, tabulated in Table S5-1.

Table S5-1. Doubling times and Max OD_{600} of recoded genome lineage

Strain	Doubling time (min.)	Doubling time standard deviation	Max OD_{600}	Max OD_{600} standard deviation
MG1655	47	1	1.09	0.01
EcNR2	47	1	1.04	0.03
rEc1	51	2	0.94	0.01
rEc2	49	1	1.02	0.03
rEc3	49	2	1.09	0.02
rEc4	48	1	1.03	0.01
rEc5	49	1	0.90	0.03
rEc6	50	1	0.92	0.02
rEc7	48	1	1.06	0.02
rEc8	49	1	1.00	0.02
rEc9	50	1	1.01	0.01
rEc10	49	1	1.02	0.02
rEc11	47	2	1.02	0.01
rEc12	51	1	1.03	0.02
rEc13	52	2	1.07	0.02
rEc14	49	3	1.05	0.00
rEc21	46	2	1.08	0.01
rEc22	49	2	1.05	0.01
rEc23	48	1	1.05	0.02
rEc24	48	1	0.99	0.01
rEc25	45	2	1.04	0.02
rEc26	48	2	1.10	0.01
rEc27	50	3	1.03	0.01
rEc28	49	1	1.00	0.01
rEc29	44	1	1.01	0.01
rEc30	53	3	1.01	0.02
rEc31	48	1	1.13	0.01
rEc32	49	1	1.06	0.02
Conj1	54	3	1.03	0.04
Conj2	52	2	1.09	0.04
Conj3	71	0	0.59	0.08
Conj4	46	1	1.19	0.02

Table S5-1 (Continued).				
Conj5	54	1	1.10	0.05
Conj6	57	2	1.07	0.04
Conj7	52	4	1.01	0.03
Conj8	47	2	1.05	0.01
Conj11	86	19	0.73	0.16
Conj12	49	1	1.13	0.02
Conj13	46	2	1.10	0.03
Conj14	47	2	1.14	0.01
Conj15	90	31	0.78	0.32
Conj16	49	1	1.05	0.13
Conj17	50	1	1.02	0.03
Conj18	56	3	1.02	0.01
Conj19	54	1	1.03	0.04
Conj20	50	2	1.01	0.01
Conj21[a]	54	4	1.16	0.05
Conj22	55	0	1.06	0.01
Conj23	74	5	1.05	0.02
Conj24	75	5	1.06	0.06
Conj25	56	3	0.97	0.03
Conj26	52	3	1.00	0.02
Conj27	55	1	1.11	0.02
Conj28	66	3	1.01	0.01
Conj29	63	0	0.96	0.03
Conj30	68	4	0.99	0.04
Conj30.5	90	8	0.62	0.13
C321.ΔA[a]	75	1	0.95	0.01

[a] Conj21 and C321.ΔA growth curves were performed separately from the others

Whole-genome sequencing

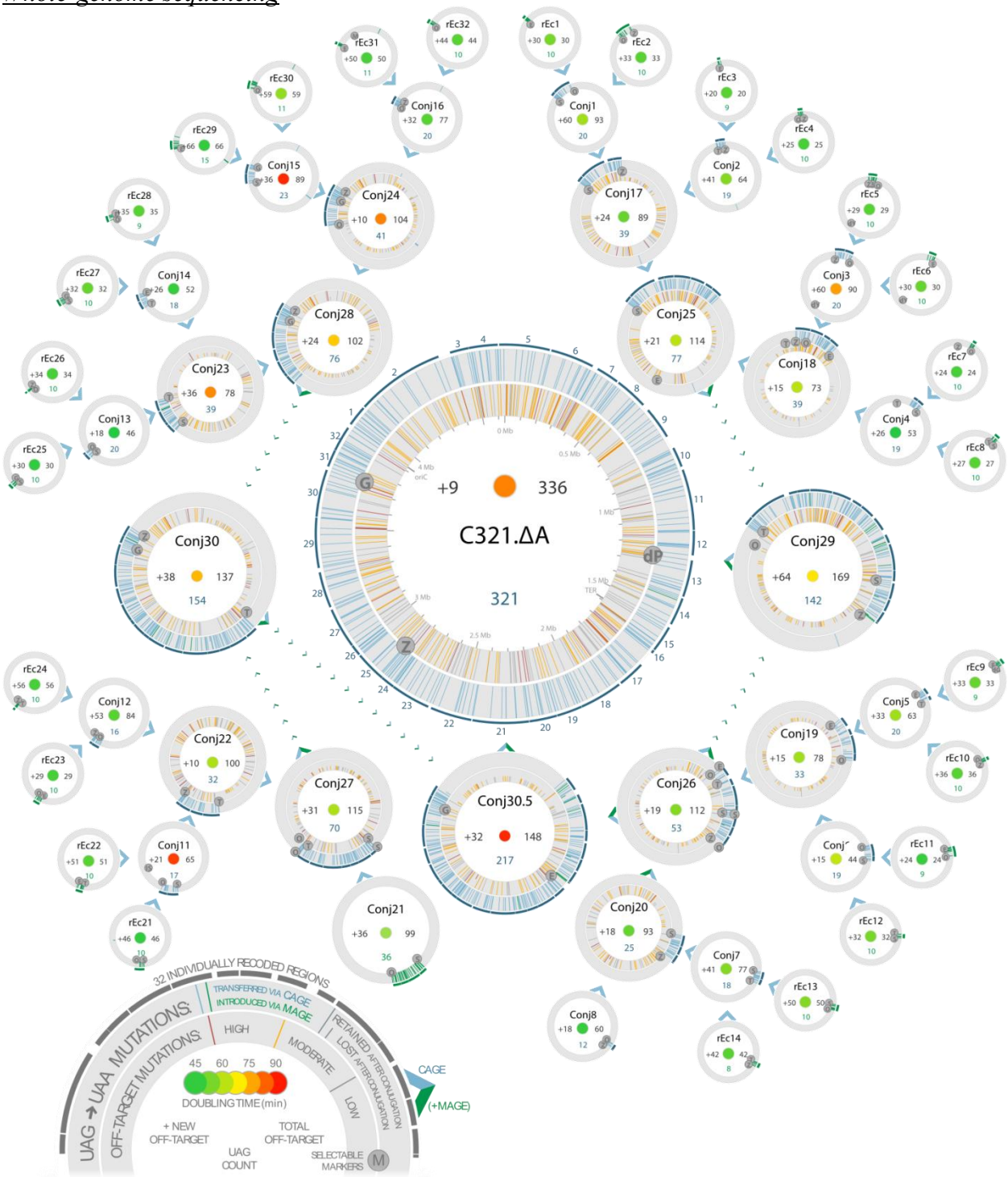


Figure S5-3. Construction and analysis of C321.ΔA. The genome was conceptually divided into 32 segments, each containing 10 UAG codons. MAGE (*13*) was used to convert all 10 UAG codons to the synonymous UAA codon in each segment across 32 parallel strains, and CAGE (*11*) was used to hierarchically assemble recoded genome segments into a fully recoded chromosome. Blue arrows point from each strain to its conjugal progeny; blue and green arrows

Figure S5-3 (Continued). indicate when MAGE was used to convert remaining UAG codons. Strain names (top), total UAGs removed (bottom, Table S5-3), new off-target mutations (left), total off-target mutations (right, Table S5-2), and doubling times (green to yellow to red gradient indicates increasing doubling times; Table S5-1) are reported at the center of each genome. Radial lines in each genome indicate the positions of mutations. The outer circle shows all UAG codons that have been replaced with UAA (green indicates UAG→UAA introduced *via* MAGE and blue indicates UAG→UAA transferred *via* CAGE). The inner circle indicates all off-target mutations acquired during recoded genome construction (color indicates mutation severity according to snpEFF (*34*): gray = low, orange = medium, and red = high). Full lines are mutations that were transferred by CAGE, and half lines are mutations that were lost during conjugation. Approximate positions of conjugal crossovers can be inferred based on which mutations were transferred. A complete list of mutations can be found in Table S5-4. Gray circles indicate positions of selectable markers immediately before conjugation (O = kan^R-oriT, T = *tolC*, G = *galK*, M = *malK*, S = $spec^R$, E = $gent^R$, Z = zeo^R, dP = Δ*prfA*, IS = *tolC*::IS5). In cases where marker symbols overlapped, they were repositioned for clarity. Strains rEC15 through rEC20 are not included because Conj21 was constructed entirely via CoS-MAGE.

Overview of genome sequencing: Genome sequencing confirmed that all 321 known UAGs have been removed from its genome and that 355 additional mutations were acquired during strain construction (1E-8 mutations/bp/doubling over ~7340 doublings; Figure S5-3, Table S5-2). Only 51 of these unintended mutations were predicted to be highly disruptive by snpEFF (Table S5-3) (*34*), providing a tractable number of alleles that could be reverted *via* MAGE to potentially improve fitness. Only one *bona fide* IS element transposition event (IS5 in Conj11) and one putative rearrangement (Conj20) were observed, suggesting that structural variants are rare. We also sequenced and characterized the complete CAGE lineage, and observed that the intermediate strains exhibited varying fitness (Figure 5-2), as expected for mutator (*i.e.*, Δ*mutS*) strains forced through monoclonal bottlenecks (*57*). Notably, the fitness defects in Conj3, Conj11, Conj15, Conj23, and Conj24 were mitigated in their conjugal progeny even though the UAG→UAA mutations from these strains were inherited (Figure S5-3 and Table S5-4). This suggests that off-target mutations likely caused the observed fitness defects, and that CAGE can eliminate deleterious mutations by preferentially selecting healthy alleles from one parent. Sequencing indicated that MAGE cycling in the rEc strains resulted in an average of 37.4 unintended mutations per strain after ~372 doublings (2E-8 mutations/bp/doubling). Across the entire lineage, we observed only 39 putative MAGE oligonucleotide synthesis errors and 6 putative oligonucleotide mistargeting events resulting in mutations at homologous sequences elsewhere in the genome, rather than the desired target. Therefore, MAGE oligonucleotides do not appear to be a major cause of mutagenesis. Of the remaining 2,225 off-target mutations in the lineage (Table S5-3), 92% were transitions (A•T→G•C and G•C→A•T) (*58*), suggesting that MutS inactivation underlies most of the unintended mutagenesis (*58*).

Off-target mutations: There are many ways that unintended mutations occur. Mismatch repair deficiency probably accounted for the majority of the 2270 off-target mutations across all 69 strains that were sequenced. Additionally, MAGE oligos can introduce off-target mutations *via* recombination. Oligos that contain chemical synthesis errors can introduce off-target mutations near their desired UAG→UAA mutation, and oligos can mistarget to homologous sequences elsewhere in the genome.

Summary of SNPs: The number of mutations introduced into each strain of the C321.ΔA lineage is summarized in Table S5-3, including the breakdown of SNP severity according to snpEFF (Table S5-24) (*34*). All mutations and their predicted severity are tabulated in Table S5-4. This information could be used to identify off-target mutations that were responsible for the transiently reduced fitness of Conj3, Conj11, Conj15, Conj23, and Conj24, but that were not propagated inherited *via* CAGE. Furthermore, by comparing the severity and location of off-target mutations in C321.ΔA, candidate alleles could be identified for reversion in an attempt to ameliorate its reduced fitness.

Table S5-3 is attached separately, and contains a summary of SNP types per strain (UAG→UAA mutations, SNPs originating from off-target mutagenesis, SNPs due to oligo-synthesis errors and MAGE oligo mistargeting) and the number of SNPs transferred by each strain during CAGE. This table also summarizes the number of SNPs in each strain according to snpEFF severity (*34*). The categories are as follows:

- SAMPLE = Name of the sample.
- STRAIN_NUM = Identification number for this strain.
- NEW_OT_OLIGO = Number of new off-target SNPs in this strain that fall in regions targetted by MAGE oligos.
- NEW_OT = Total number of new off-target SNPs in this strain.
- NEW_OT_MT = Number of new off-target SNPs in this strain that fall into regions with significant homology to MAGE oligos (indicative of MAGE mistargetting).
- NEW_OT_TS = Number of new off-target SNPs in this strain that are transitions.
- NEW_OT_NOT_OLIGO_TS = Number of new off-target SNPs in this strain that are transitions and not in regions targetted by MAGE oligos.
- NEW_OT_NOT_XFER = Number of new off-target SNPs in this strain that are transferred to the child strain via CAGE.
- TOTAL_OT = Total number of off-target mutations in this strain. TOTAL_MT = Total number of mutations in this strain that fall into regions with significant homology to MAGE oligos (indicative of MAGE mistargetting).
- NEW_AMBER = Number of new UAG to UAA SNPs in this strain.
- TOTAL_AMBER = Total number of UAG to UAA SNPs in this strain.
- EFF_NONE* = Number of SNPs in this strain with no known effect on genic regions.
- EFF_LO* = Number of SNPs in this strain with an effect characterized by snpEFF as "low".
- EFF_MED* = Number of SNPs in this strain with an effect characterized by snpEFF as "moderate".
- EFF_HI* = Number of SNPs in this strain with an effect characterized by snpEFF as "high".

* In cases where SNPs have multiple effects, the highest is reported.

Table S5-24. Summary of snpEFF types

High	START_LOST FRAME_SHIFT STOP_GAINED STOP_LOST
Moderate	NON_SYNONYMOUS_CODING CODON_CHANGE CODON_INSERTION CODON_CHANGE_PLUS_CODON_INSERTION CODON_DELETION CODON_CHANGE_PLUS_CODON_DELETION
Low	SYNONYMOUS_START NON_SYNONYMOUS_START START_GAINED SYNONYMOUS_CODING SYNONYMOUS_STOP

Table S5-4 is attached separately, and contains an exhaustive list of all called SNPs per strain, including those that passed the initial Freebayes filtering but not the more stringent downstream filters. The categories are as follows:

- SAMPLE = Name of the strain.
- POS = Chromosome name and position.
- seqnames = Chromosome name.
- start = SNP start position.
- end = SNP end position.
- width = Width of event in bases.
- REF = Reference allele.
- ALT = Alternate allele(s).
- QUAL = SNP quality metric.
- NS = Number of samples in which the SNP was called.
- DP = Total depth across all samples.
- AC = Total number of alternate alleles in called genotypes.
- AF = Estimated allele frequency in the range (0,1].
- RO = Reference allele observations.
- AO = Alternate allele observations.
- AB = Allele balance ratio.
- RUN = Run length (the number of consecutive repeats of the alternate allele in the reference genome).
- DPRA = Alternate allele depth ratio (ratio between ALT SNP calls and WT SNP calls for a given allele and strain)

- TYPE = The type of allele (snp, mnp, ins, del, or complex).
- LEN = Allele length.
- MQM = Mean mapping quality of observed alternate alleles.
- MQMR = Mean mapping quality of observed reference alleles.
- PAIRED = Proportion of observed alternate alleles which are supported by properly paired read fragments.
- PAIREDR = Proportion of observed reference alleles which are supported by properly paired read fragments.
- EFF = Effect string from snpEFF.
- EFF_TYPE = Effect types.
- EFF_SEV = Effect severities.
- EFF_FUNC = Effect functional class.
- EFF_CODON = Effect codon data, if SNP changes a codon.
- EFF_AA = Effect amino acid data, if SNP changes an amino acid.
- EFF_GENE = Gene(s) which this SNP affects.
- EFF_SEV_HIGHEST = The highest severity of all effects for this SNP.
- S_GT = Sample genotype.
- S_GQ = Genotype quality, the Phred-scaled probability of the called genotype.
- S_DP = Sample read depth.
- S_RO = Sample read observations.
- S_QR = Sum of quality of the alternate observations.
- S_QA = Sum of quality of the reference observations.
- S_AO = Alternate allele observation count.
- GT.A = If heterozygous, WT/ALT status 1.
- GT.B = If heterozygous, WT/ALT status 2.
- HET = Is this SNP called as ‘heterozygous’ (see supplemental methods).
- NC = Is this SNP not called for this genome.
- CALL = Call status (0 for WT, 1+ for ALT).
- VAR = Is this SNP not WT.
- DISPLAY_NAME = Display name of the sample.
- PARENT = Parent strains for this strain.
- CHILD = Child strains for this strain.
- STRAIN = Strain name.
- STRAIN_TYPE = Strain type.
- STRAIN_ID = Strain ID.
- IN_OLIGO = Is this SNP in a region targetted by a MAGE oligo
- AMBER = Is this an UAG to UAA SNP?
- AMBER_COUNT = Number of UAG to UAA mutations made in this SNP.
- IN_CHILD = Number of child strains that received this SNP from this strain.
- IN_PARENT = Number of parent strains that passed this SNP to this strain.
- NO_CALL = Was this SNP not called for this strain?
- NC_COUNT = Number of strains in which this SNP was not called WT/ALT.
- C_COUNT = Number of strains in which this SNP was called WT/ALT.
- NC_PCT = Percentage of strains in which this SNP could not be called.

- INSUFF_CALLS = Flag for whether or not this SNP was called in too few samples.
- AO_TOTAL = Total number of alternate observations across all alternate alleles.
- INSUFF_READS = Flag for whether or not this SNP had too few good quality mapped reads across all samples.
- INSUFF_SAMPLES = Flag for whether or not this SNP was called in too few samples.
- BAD = Flagged if this SNP had either insufficient calls, reads, or called samples.
- ANCESTRAL = Does this SNP occur in MG1655 or EcNR2?
- FILTER = Does this SNP match all the criteria described in the supplemental SNP filtering methods?
- DISPLAY = Should this SNP be displayed in Figure 5-2? (FILTER + !ANCESTRAL)
- TS = Is this SNP a transition mutation (A→G, G→A, C→T, or T→C)?

Chemical synthesis errors: We detected 39 off-target mutation events in regions targeted by MAGE oligos in the strains that underwent extensive MAGE cycling (rEc strains and C321). Of these, 16 were mismatches, 23 were deletions, and 0 were insertions. A subset of these mutations may be caused by spontaneous mutagenesis (Δ*mutS*).

MAGE oligo mistargeting: We used blastn (default parameters, http://blast.ncbi.nlm.nih.gov/) to identify 31 MAGE oligos in regions of the genome that shared homology with the intended oligo site (Table S5-25).

Table S5-25. Summary of blastn results for potential MAGE oligo mistargeting regions

Oligo ID	Avg. align length	Avg. nt identity	Number of alignments[a]	Total align length
ascB	28.10112	100.0	89	2501
aslB	91.00000	95.6	1	91
b0299	76.80000	98.4	5	384
b0361	58.85714	97.7	7	412
b1228	91.00000	92.3	1	91
b1402	56.25000	98.2	8	450
b1578	56.25000	98.2	8	450
b1996	57.25000	98.3	8	458
b2860	57.50000	98.3	8	460
b3045	59.00000	97.2	8	472
b4273	57.14286	98.3	7	400
b4283	54.66667	96.6	3	164
eaeH	65.00000	98.5	5	325
hda	28.00000	100.0	1	28
hokE	61.00000	95.9	2	122
insB	88.00000	89.8	7	616
rcsC	35.63333	95.9	60	2138
rhsA	77.00000	94.8	2	154
tfaE	90.00000	95.6	1	90
tfaS	90.00000	94.4	1	90
tra5_1	78.00000	98.0	5	390

Table S5-25 (Continued).

tra5_2	78.16667	98.3	6	469
tra5_3	76.83333	98.3	6	461
tra5_4	77.66667	97.9	6	466
yafF	35.33333	100.0	3	106
yafL	34.84483	96.5	58	2021
yahH	44.83333	91.1	12	538
ygeP	45.75000	99.2	8	366
yghQ	48.00000	98.5	11	528
yjjV	37.32979	96.4	94	3509
yrhA	76.25000	97.5	4	305

[a] Number of times each oligo aligns at genomic locations other than the desired target location

There were 61 total unique mutations in the regions identified by BLAST. Of the 44 that passed filter, 4 were already present in EcNR2, 16 were on-target UAG→UAA mutations, and 28 were potentially caused by oligo mistargeting. Because some mutations were found in multiple strains, we detected 32 total off-target mutations that shared homology with at least one MAGE oligo. To verify putative mistargeting events, we identified all oligos that satisfied the following requirements: (i) the oligo had been MAGE cycled in the mutated strain in question and (ii) the oligo was homologous to the region in which the mutation occurred. According to these criteria, there were only 6 likely mistargeting events (Table S5-26).

- There were 5 *bona fide* mistargeting events—putative mistargeting resulted in mutations that matched the oligo sequence.
- There was 1 putative mistargeting event—putative mistargeting resulted in mutation that may have been caused by a chemical synthesis error in the MAGE oligo.
- There were 26 putative false positives:
    - There were 7 putative synthesis errors from proper MAGE oligo targeting that were identified as off-target homologies for other oligos (some oligos that target repetitive elements share similar sequences to each other).
    - There were 9 putative spontaneous mutations (mutations in mistargeting homology regions for MAGE oligos that were not used in the mutated strain).
    - There were 10 heterozygous mutations toward the b1228 oligo sequence in strains that had never been exposed to this oligo (probably an artifact of binary heterozygous SNP calling).

Off-target structural variants: With the possible exception of the Conj 20 and Conj 26 rearrangement described above, we found few instances of structural variants that could be caused by CAGE. This analysis is based on Pindel (*35*) and Breakdancer (*36*) output, which primarily identified the known marker insertion sites. Table S5-27 and Table S5-28 report all uncharacterized Pindel breakpoint events and all complete structural events, respectively. All reported events have at least 20 split reads supporting them. Additionally, Table S5-29 reports all high quality Breakdancer events that are supported by a minimum of 8 reads and have a quality score of at least 20. False positives and false negatives were observed in output from both Pindel and Breakdancer. Therefore, as described in the methods section each structural variant must be confirmed by hand using samtools tview <http://samtools.sourceforge.net/tview.shtml> (*38*).

CAGE removes deleterious alleles: We observed several cases in which CAGE improved fitness in conjugal progeny by allowing preferential inheritance of healthy alleles from one parental strain. This effect is most pronounced during the early stages of CAGE in which the recoded segment is small, and the conjugal junctions are less constrained. However, it diminishes with increasing recoded region sizes, since random mutations become less likely to be removed by chance, and the population of desired genotypes becomes smaller (Table S5-16) (*53*).

Generating C321.ΔA sequence annotation file (genbank format): We generated an annotated sequence file in Genbank format for C321.ΔA using custom software. This process required us to scrutinize the above SNP and structural variant analysis at a deeper level and resulted in accepting an additional 19 SNPs and 2 deletions that had been previously identified by Freebayes or Pindel or Breakdancer, but which had been triaged based on heuristics intended to remove false positives.

The software takes as input:

- MG1655 reference Genbank with accession number NC_000913 from NCBI
- List of UAG positions in MG1655 (Table S5-34).
- List of manual fixes which include cassette insertions and deletions (*e.g.* delete *prfA*, insert lambda prophage), as well as the 2 structural variations and 19 SNPs that were hand-validated as described above (Table S5-35)
- List of remaining off-target variants as called by Freebayes (Table S5-36)

Our software applies these changes and outputs an annotated file in Genbank format. We then realigned the C321.ΔA fastQ sequencing reads to this genbank file, and re-ran the variant-calling pipeline to identify any discrepancies. By repeating this process iteratively, we were able to identify variants that were previously filtered out due to insufficient evidence based on the MG1655 reference sequence.

Finally, we wrote another custom script to convert our Genbank file into the .sqn submission format required by NCBI. This was done by generating a five-column table format representing the feature annotations which is then fed into the NCBI script tbl2asn. This script performs an additional layer of validation on the annotated sequence according to well-established biological rules, and generates the submission file to be sent to NCBI. The sequence and annotation were submitted to NCBI for release at time of publication.

Current technologies are inadequate:

Modern next-generation sequencing (*e.g.* Illumina HiSeq) now allows for dozens of bacterial strains to be sequenced simultaneously and in a matter of days. Despite the increasing ease of generating raw sequencing data for bacterial genomes, there are a lack of purpose-built tools to deal with this data.

Our current pipeline combines almost a dozen modular tools, many of which are designed for human genome assembly and human population genetics. We know of no existing tools that

integrate multi-step genome-scale design, short-read assembly, and SNP and structural variant detection. The development of such tools would allow for rapid iteration, testing, and troubleshooting of engineered genomes.

Additionally, while the small size of bacterial genomes makes short-read sequencing assembly relatively simple, many genomic variants remain beyond the reach of short read sequencing alone because they occur in duplicated regions (*e.g.* tRNAs, IS elements, highly paralogous genes, etc.). In many cases, short reads align to all copies of such regions with equal likelihood, making it difficult to call SNPs and structural variants in these regions. The creation of genomes with removed or diversified paralogous sequences could be combined with longer sequencing read lengths to produce correct, short-read genome sequences *via* resequencing.

H. Mass spectrometry

We hypothesized that NSAA incorporation was occurring at native UAG positions of unrecoded genomes and we thus aimed to investigate this by directly measuring this effect in the native proteome. This has not been achieved for multiple native genes and previous work relied on tagging methods (altered genes) or plasmid-based single ORFs. We chose an in-depth proteomics approach to provide an unbiased view of the native proteome. This approach comes with a few expected technological limitations of mass spectrometry. Currently, no single proteomics method, or combination of methods, allows for 100% sequence coverage of all proteins. Our shotgun methods, which are slightly better than recent reports (*60*), have an inherent bias towards the detection of higher abundance proteins. We detected over 1,000 proteins (~1/4th of the proteome) and only 40 to 60 of the proteins detected were UAG containing ORFs. The major reason we do not observe more NSAA peptides is that the majority of UAG ORFs are lower in abundance and not in the top 1,000 proteins in the cell. We therefore applied a more robust method described in the SOM that nearly doubled the number of detectable proteins and more than tripled the number of UAG ORFs detected. However, limitations such as depth of peptide covered per ORF, observable peptides with mass spectrometry compatibility properties (such as peptide length, ionization properties, and ideal trypsin cleavage sites), and non-UAG dependent termination sequences are factors that reduce the number of NSAA peptides observed. We also expect that UAG read through and NSAA incorporation would destabilize proteins and reduce their expression below detectable levels. Based on these limitations, we think our list of natural UAG suppression, which is obtained from the most technologically advanced MS methods, underrepresents the total number of natural UAG suppression events. Nevertheless, we observed a highly reproducible sampling of multiple native ORFs that tolerated two distinct types of NSAA insertions. Importantly, these events were erased from the proteome by recoding, a property we confirmed by direct observation of the proteome (Figure 5-3D and Figure S5-7). We think the native, off target NSAA insertions are relevant at any level and we confirmed that NSAA insertions occur at genes essential for viability and fitness (*e.g. mreC* and *sucB*; Figure 5-3C, Table S5-8, and Table S5-11).

Table S5-6. Summary of survey proteomic analysis of strains incorporating pAcF

Strain	OTS	NSAA	Protein ID's[a] #	UAG ORF's[b] #	UAG peptides #	FDR[c] %	FDR[d] %
C0.B*.ΔA::S	none	none	1101	49	0	1.00	1.34
C0.B*.ΔA::S	none	pAcF	1149	53	0	0.86	1.19
C0.B*.ΔA::S	pEVOL-pAcF	none	1130	55	0	0.84	1.19
C0.B*.ΔA::S	pEVOL-pAcF	pAcF	1131	40	3	1.02	1.29
C314.ΔA::S	none	none	1139	60	0	0.85	1.22
C314.ΔA::S	none	pAcF	1138	64	0	0.81	1.22
C314.ΔA::S	pEVOL-pAcF	none	1042	62	0	0.97	1.31
C314.ΔA::S	pEVOL-pAcF	pAcF	1006	55	0	0.96	1.34

[a] Protein ID statistics from Yale Protein Expression Database (YPED)
[b] Identified by searching UAG only DB, retrieved from MASCOT, 5 % False Discovery Rate (FDR)
[c] Peptide matches above identity threshold (YPED)
[d] Peptide matches above homology or identity threshold

Table S5-7. Summary of in-depth proteomics of strains incorporating pAcF

Strain	OTS	NSAA	Protein ID's[a] #	UAG ORF's[b] #	UAG peptides #	FDR[c] %	FDR[d] %
C0.B*.ΔA::S	pEVOL-pAcF	pAcF	1814	137	9[e]	0.87	2.45
C314.ΔA::S	pEVOL-pAcF	pAcF	1803	163	0	1.05	2.58

[a] Protein ID statistics from YPED
[b] Identified by searching UAG only DB, pulled from MASCOT
[c] Peptide matches above identity threshold (YPED)
[d] Peptide matches above homology or identity threshold
[e] 11 suppressed UAG codons (two UAG codons each in SucB and YbjK peptides)

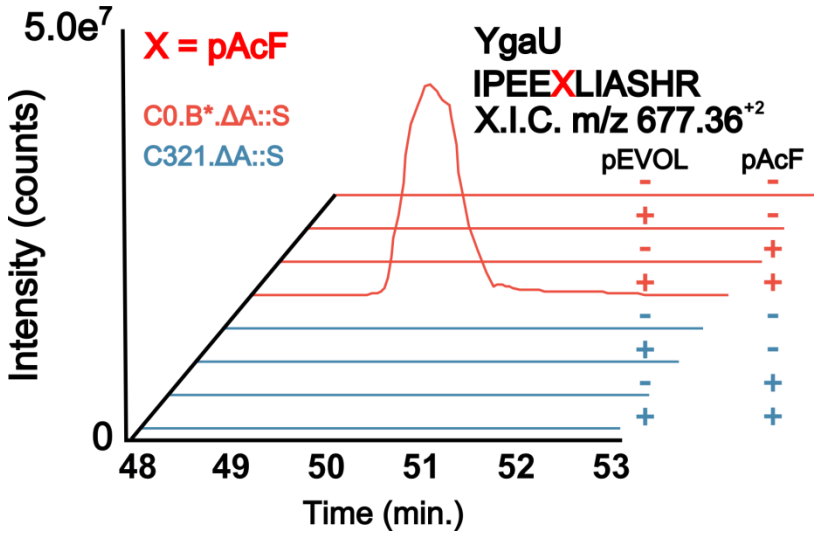


Figure S5-7. Extracted ion chromatograms are shown for pAcF incorporation into the YgaU peptide. Peptides containing pAcF were only observed in C0.B*.ΔA::S, and not in C321.ΔA::S, when pEVOL-pAcF was induced and pAcF was supplemented.

Table S5-8. Summary of identified pAcF containing peptides

Protein	Peptide sequence[a]	Experimental MW	Calculated MW	Delta mass ppm	MASCOT Ion score
FrmR	XLNLLPY	920.5022	920.5007	1.6	16.47
SucB	LLLDVXXFK	1224.6668	1224.6794	10.3	26.44
YbjK	VAGXXISFR	1126.5826	1126.5811	1.3	55.27
MarA	FLHPLNHYNSXLK	1671.8607[b]	1670.8569	600.8[b]	33.03
SpeG	TPGQTLLKPTAQXH	1579.8371	1579.8358	0.8	67.53
YgaU	IPEEXLIASHR	1352.7096	1352.7088	0.6	88.36
LuxS	LQELHIXSVNYLHN	1767.8927	1767.8944	1.0	62.2
LldD	GNAAXSFAPPHPNPLPQGEGTVR	2402.1769	2402.1767	0.1	54.39
IlvA	LMXPLFLR	1077.6050	1077.6045	0.5	30.95

[a] X = pAcF
[b] ^{13}C isotope

Table S5-9. Summary of all identified proteins with pAcF incorporation at UAG codon(s)

Protein	Description[a]	**C0**.ΔA::S + **pEVOL** + **pAcF**	C314.ΔA::S + **pEVOL** + **pAcF**
FrmR	Regulator protein that represses frmRAB operon	+	-
SucB	Dihydrolipoyltranssuccinase	+	-
YbjK	Predicted DNA-binding transcriptional regulator	+	-
MarA	DNA-binding transcriptional dual activator of multiple antibiotic resistance	+	-
SpeG	Spermidine N1-acetyltransferase	+	-
YgaU	Predicted protein	+	-
LuxS	S-ribosylhomocysteine lyase	+	-
LldD	L-lactate dehydrogenase, FMN-linked	+	-
IlvA	Threonine deaminase	+	-

[a] Gene functions were referenced from <http://www.ecocyc.org> (*45*).

Supplemental Information: Phosphoserine

Table S5-10a. Summary from the proteomic analysis of the TiO_2 enriched fraction of strains containing Sep-TECH

Strain[a]	OTS	NSAA	Protein ID's[b] #	UAG ORF's[c] #	UAG peptides #	FDR[d] %	FDR[e] %
EcNR2.ΔserB	SepRS/tRNA^{Sep}	Sep	313	17	0	1.15	3.26
EcNR2.ΔserB	SepRS/tRNA^{Sep}	Sep	292	21	0	0.55	1.76
C0.B*.ΔA::S.ΔserB	SepRS/tRNA^{Sep}	Sep	325	23	6	0.64	1.93
C0.B*.ΔA::S.ΔserB	SepRS/tRNA^{Sep}	Sep	249	21	5	0.82	2.42
C13.ΔA::S.ΔserB	SepRS/tRNA^{Sep}	Sep	188	20	4	1.63	3.05
C13.ΔA::S.ΔserB	SepRS/tRNA^{Sep}	Sep	314	16	5	0.92	1.88
C321.ΔA::S.ΔserB	SepRS/tRNA^{Sep}	Sep	227	12	1[f]	1.25	2.45
C321.ΔA::S.ΔserB	SepRS/tRNA^{Sep}	Sep	335	20	1[f]	0.90	2.65

[a] All strains harbored pKD-SepRS-EFsep and pSepT (Sep OTS) and were supplemented with Sep
[b] Protein ID statistics from Yale Protein Expression Database (YPED), results from biological replicates are listed separately
[c] Identified by searching UAG only DB, retrieved from MASCOT, 5 % False Discovery Rate (FDR)
[d] Peptide matches above identity threshold (YPED)
[e] Peptide matches above homology or identity threshold
[f] Carryover levels observed (source of carryover from prior MS run: C13.ΔA::S.ΔserB

We observed only a single NSAA peptide (resulting from native UAG suppression) in two samples from C321.ΔA::S.ΔserB. We loaded 4µg of peptides for these LC-MS runs and followed each run with 2 different types of blank runs designed to clean the LC column. However, we still observed a small amount of carryover, after the two blanks that introduced a small amount of a phosphoserine peptide into the C321.ΔA::S.ΔserB sample from the previous C13.ΔA::S.ΔserB run. We re-ran the set of 4 samples at 1ug loads with the same blank runs and saw no detectable carryover (i.e. this eliminated the detection of the single carryover phosphopeptide from the C321.ΔA::S.ΔserB sample).

Table S5-10b. Summary from the proteomic analysis of the TiO_2 enriched fraction of strains containing Sep-TECH

Strain[a]	OTS	NSAA	Protein ID's[b] #	UAG ORF's[c] #	TAG peptides #	FDR[d] %	FDR[e] %
EcNR2.ΔserB	SepRS/tRNA^{Sep}	Sep	249	7	0	0.82	2.42
C0.B*.ΔA::S.ΔserB	SepRS/tRNA^{Sep}	Sep	202	9	3	0.29	2.43
C13.ΔA::S.ΔserB	SepRS/tRNA^{Sep}	Sep	188	12	2	1.63	3.05
C321.ΔA::S.ΔserB	SepRS/tRNA^{Sep}	Sep	198	6	0	0.88	1.96

[a] All strains harbored pKD-SepRS-EFsep and pSepT (Sep OTS) and were supplemented with Sep
[b] Protein ID statistics from Yale Protein Expression Database (YPED), results from biological replicates are listed separately
[c] Identified by searching UAG only DB, retrieved from MASCOT, 5 % False Discovery Rate (FDR)
[d] Peptide matches above identity threshold (YPED)
[e] Peptide matches above homology or identity threshold

Table S5-11. Summary of Sep-containing peptides identified by proteomics from two biological replicates each

Protein	Peptide sequence[a]	Experimental MW	Calculated MW	Delta mass ppm	MASCOT Ion score
LuxS	LQELHIXSVNYLHN	1745.8160	1745.8138	1.3	45.52
SpeG	TPGQTLLKPTAQXH	1557.7579	1557.7552	1.7	76.26
RlpA	LQTEAQLQSFITTAQXR	2000.9606	2000.9568	1.9	58.17
MreC	APGGQXWR	937.3808	937.3807	0.1	39.2
Nei	FGAXVEINR	1071.4735	1071.4750	1.4	53.06
LldD	GNAASXFAPPHPNPLPQGEGTVR	2380.1013	2380.0961	2.2	43.21
YhbW	EELLGXCVLTR	1355.6204	1355.6156	3.5	39.57
LpxK	LLTQLTLLASGNXLR	1678.9069	1678.9019	3.0	35.78

[a] X = Sep

Table S5-12. Summary of all identified proteins with Sep incorporation at an amber stop codon

Protein	Description[a]	EcNR2. ΔserB + OTS[b]	C0.B*.ΔA::S. ΔserB + OTS[b]	C13.ΔA::S. ΔserB + OTS[b]	C321.ΔA::S. ΔserB + OTS[b]
LuxS	S-ribosylhomocysteine lyase	-	+	+	+[c]
SpeG	Spermidine N1-acetyltransferase	-	+	+	-
RlpA	Septal ring protein, suppressor of prc, minor lipoprotein	-	+	+	-
MreC	Cell wall structural complex MreBCD transmembrane component MreC	-	+	-	-
Nei	Endonuclease VIII/ 5-formyluracil/5-hydroxymethyluracil DNA glycosylase	-	+	+	-
LldD	L-lactate dehydrogenase, FMN-linked	-	+	+	-
YhbW	Predicted enzyme	-	+	-	-
LpxK	Lipid A 4'kinase	-	+	-	-

[a] Gene functions were referenced from <http://www.ecocyc.org> (*45*).
[b] OTS = pKD-SepRS-EFsep and pSepT
[c] Contaminant levels observed (source of contamination from prior MS run: C13.ΔA::S.ΔserB)

I. NSAA incorporation

One of the main goals of reassigning the genetic code is to provide a dedicated channel for plug-and-play incorporation of NSAAs. To this end, we have created a robust chassis completely lacking UAG function, which is capable of accepting orthogonal aaRS/tRNA pairs. We have shown that the only known strategy to completely abolish UAG function is to remove all instances of UAG from the genome and then delete RF1. We have verified previous reports (*17, 54*) that the RF2 variant (frameshift removed, T246A, A293E) can permit RF1 deletion, but also weakly terminates at UAG codons (Figure 5-3B). Additionally, NSAA incorporation in these strains is highly toxic ((*54*) and Figure 5-3) probably because it outcompetes termination in some essential genes. This effect is particularly apparent upon outgrowth from overnight expression of pAcF and pAzF (Figure S5-5). In contrast, removing essential UAGs permits the efficient incorporation of NSAAs, but plug-and-play UAG reassignment is difficult because UAG function cannot be abolished in these strains (new UAG function must be introduced prior to RF1 deletion (*15, 17*)). Although we were able to delete RF1 without introducing a suppressor in C7.ΔA::S and C13.ΔA::S, both strains rapidly selected for efficient natural suppression. C321.ΔA::S, C321.ΔA::T, and C321.ΔA were not affected by NSAA expression. All growth curves used for this analysis are in Figure S5-17.

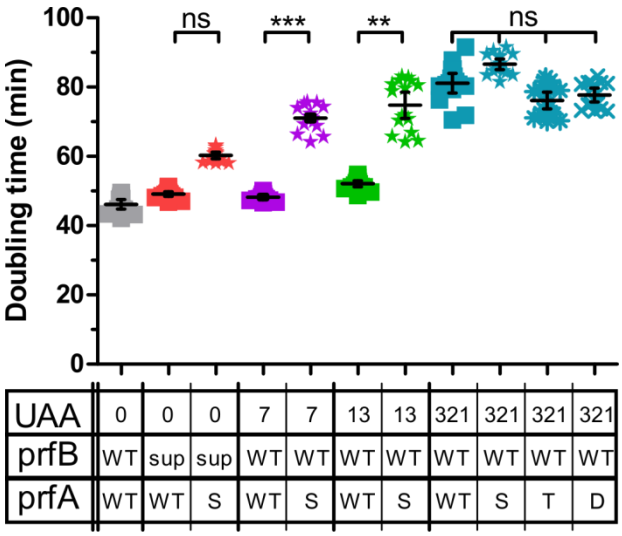


Figure S5-4. Doubling times in recoded strains +/- RF1. The number of UAG→UAA conversions are indicated by UAA. RF1 status is denoted as wt *prfA* (WT), Δ*prfA*::*spec*^{R} (S), Δ*prfA*::*tolC* (T), or Δ*prfA* (Δ). RF2 sup indicates a variant (frameshift removed, T246A, A293E) capable of suppressing lethality of RF1 deletion. While C321 has a slower growth rate than the other RF1 strains (probably due to off-target mutagenesis; see discussion in main text), RF1 deletion does not affect fitness. All other strains (C0.prfB*, C7, and C13) exhibited reduced fitness upon RF1 deletion. The gray symbols in the first column correspond to MG1655 (wild type) doubling time.

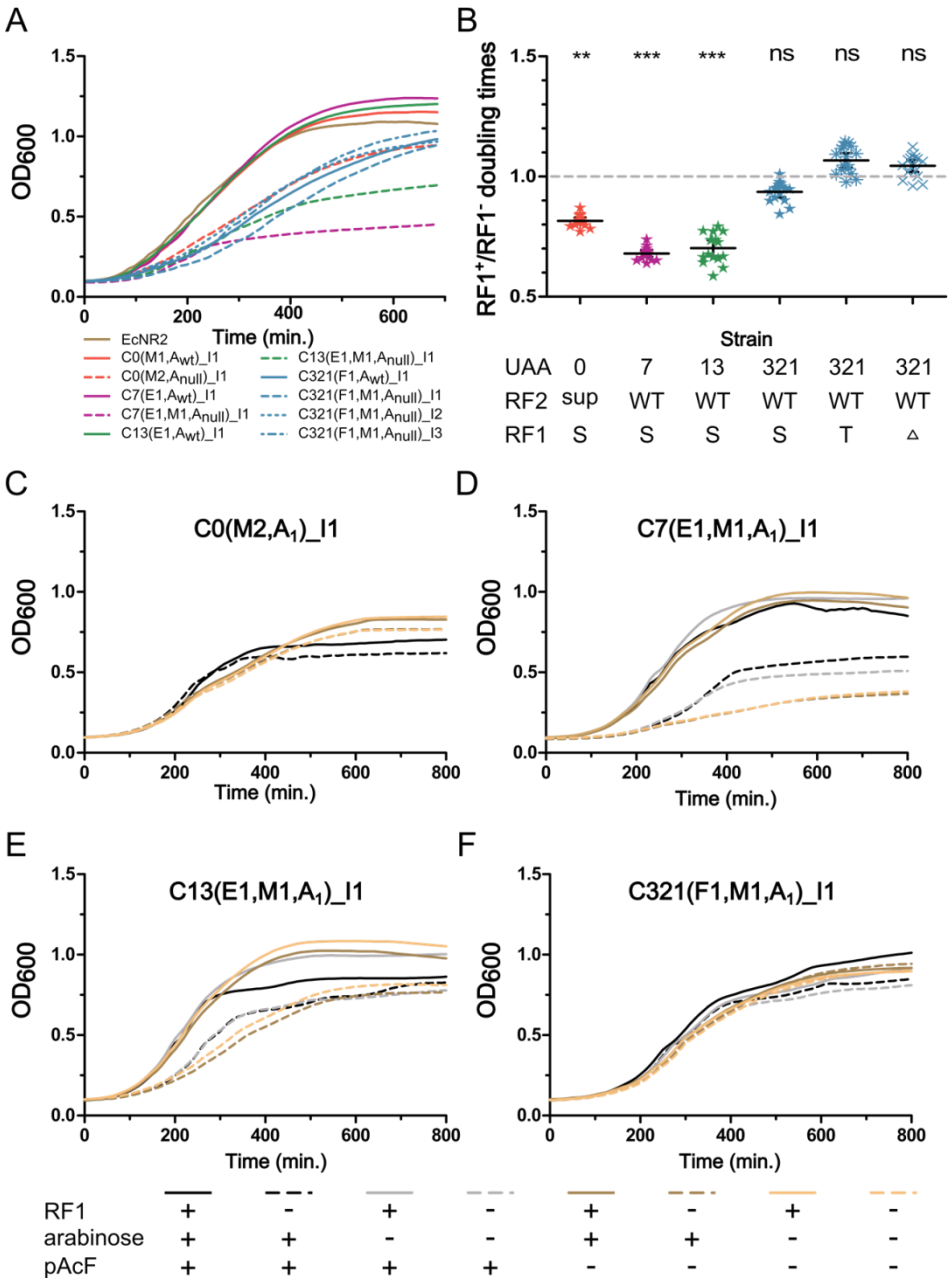


Figure S5-17. Native UAGs cause detrimental pleiotropic effects after codon reassignment. RF1 status is denoted as wt *prfA* (WT), Δ*prfA*::*spec*^{R} (S), Δ*prfA*::*tolC* (T), or Δ*prfA* (Δ). RF2 sup indicates a variant (frameshift removed, T246A, A293E) capable of suppressing lethality of RF1 deletion. **(A)** Averaged kinetic growth curves of RF1$^+$ (solid lines) and RF1$^-$ (dashed lines) strains with no UAG suppression. **(B)** Ratios of doubling times for RF1$^+$/RF1$^-$ strains with no aaRS supplemented to reassign UAG (n = 16). Statistical significance was determined using the Kruskal-Wallis test ($p < 0.0001$) followed by Dunn's multiple comparison test to compare each ratio to unity (* $p < 0.05$, ** $p < 0.01$, and *** $p < 0.001$). RF1 deletion increased doubling time and decreased maximum cell density for RF2 variants and partially recoded strains, but not for fully recoded strains. **(C-F)** Average kinetic growth curves of RF1$^+$ (solid lines) and RF1$^-$ (dashed lines) strains with pEVOL-pAcF expression and pAcF supplementation. The sense suppression of UAG impairs fitness in recoded RF2 variants (natural amino acids are incor-

Figure S5-17 (Continued). porated and impair fitness in the presence of pEVOL-pAcF even when pAcF is not supplemented) **(C)**, improves fitness in partially recoded strains **(D)** and **(E)**, and does not affect fitness in fully recoded strains **(F)**.

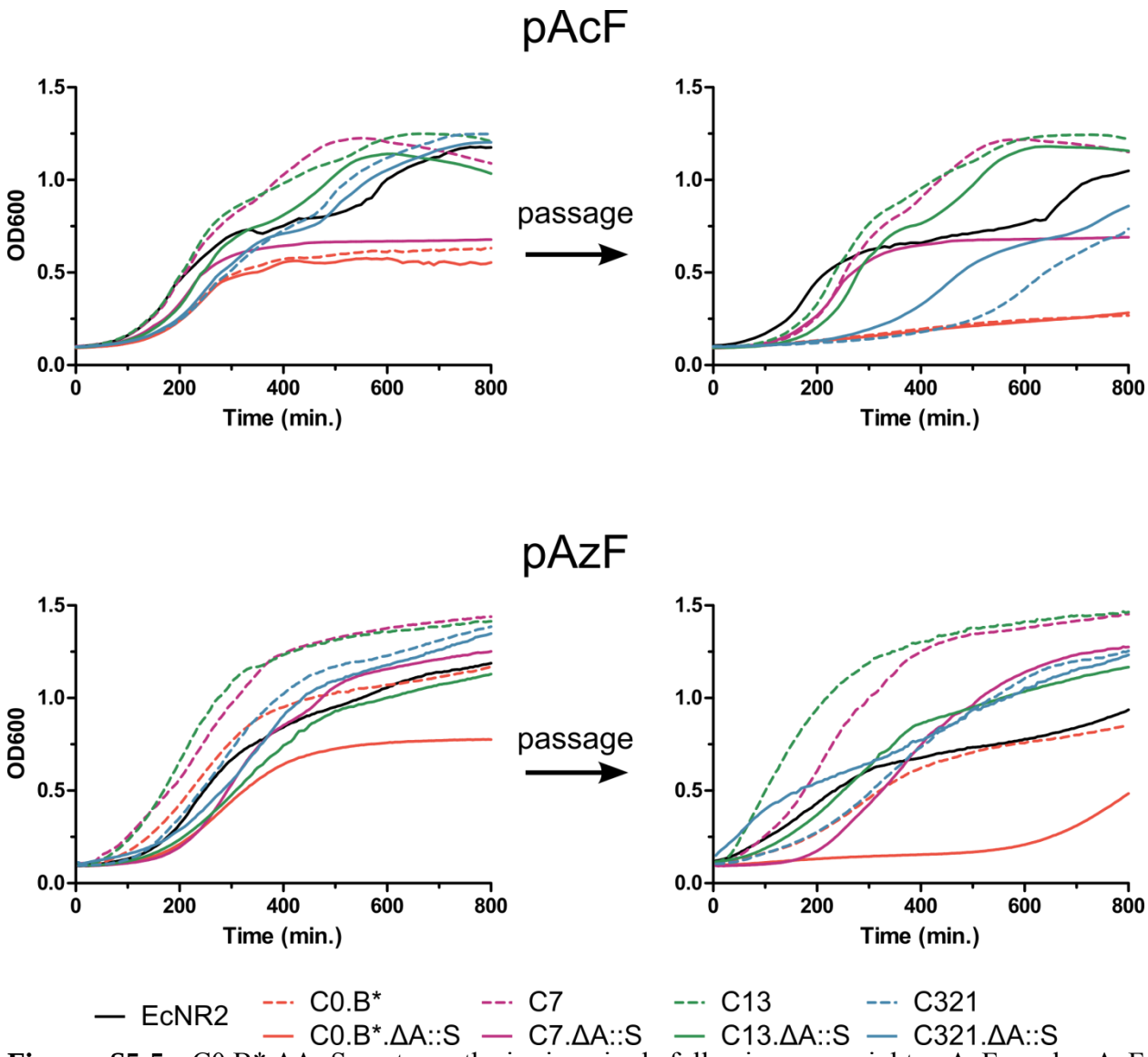


Figure S5-5. C0.B*.ΔA::S outgrowth is impaired following overnight pAcF and pAzF expression. Overnight cultures were grown in LB^L supplemented with chloramphenicol (pEVOL maintenance), arabinose (induces the aaRS), and NSAA. After 16 hours of growth, these cultures were passaged into identical media. Growth at 34°C was monitored *via* OD_{600} readings at 10-minute (pAcF) or 5-minute (pAzF) intervals using a biotek H1 plate reader.

GFP expression with UAG reassigned to p-acetylphenylalanine (pAcF)

For each recoded strain, three GFP reporters (0UAG, 1UAG, and 3UAG) were expressed in the presence and absence of pAcF, pAzF, and NapA. Figure S5-6 reports the raw fluorescence for

each strain, amino acid, and reporter gene. Therefore, fluorescence readings take into account both expression levels and cell density, which are both relevant with respect to protein overexpression. Regardless of whether this is caused by UAG recoding or off-target mutations that non-specifically increase protein production, C321.ΔA::S consistently produces the highest fluorescence on par with the wt GFP controls after 17 hours of pAcF, pAzF, or NapA expression (Figure S5-6). C0.B*.ΔA::S exhibited low fluorescence, while C7.ΔA::S and C13.ΔA::S appeared to read through UAG using canonical amino acids. C321.ΔA::S produced high levels of fluorescence, but only when the relevant NSAA was supplemented. Finally, we note that the 3UAG GFP variant produced higher fluorescence than expected in EcNR2. We verified the EcNR2 genotype, confirmed that the correct plasmid was present, and repeated the transformation of fresh pZE21G-3UAG into fresh EcNR2, but the 3UAG expression was consistently higher than the 1UAG expression in this strain for unknown reasons.

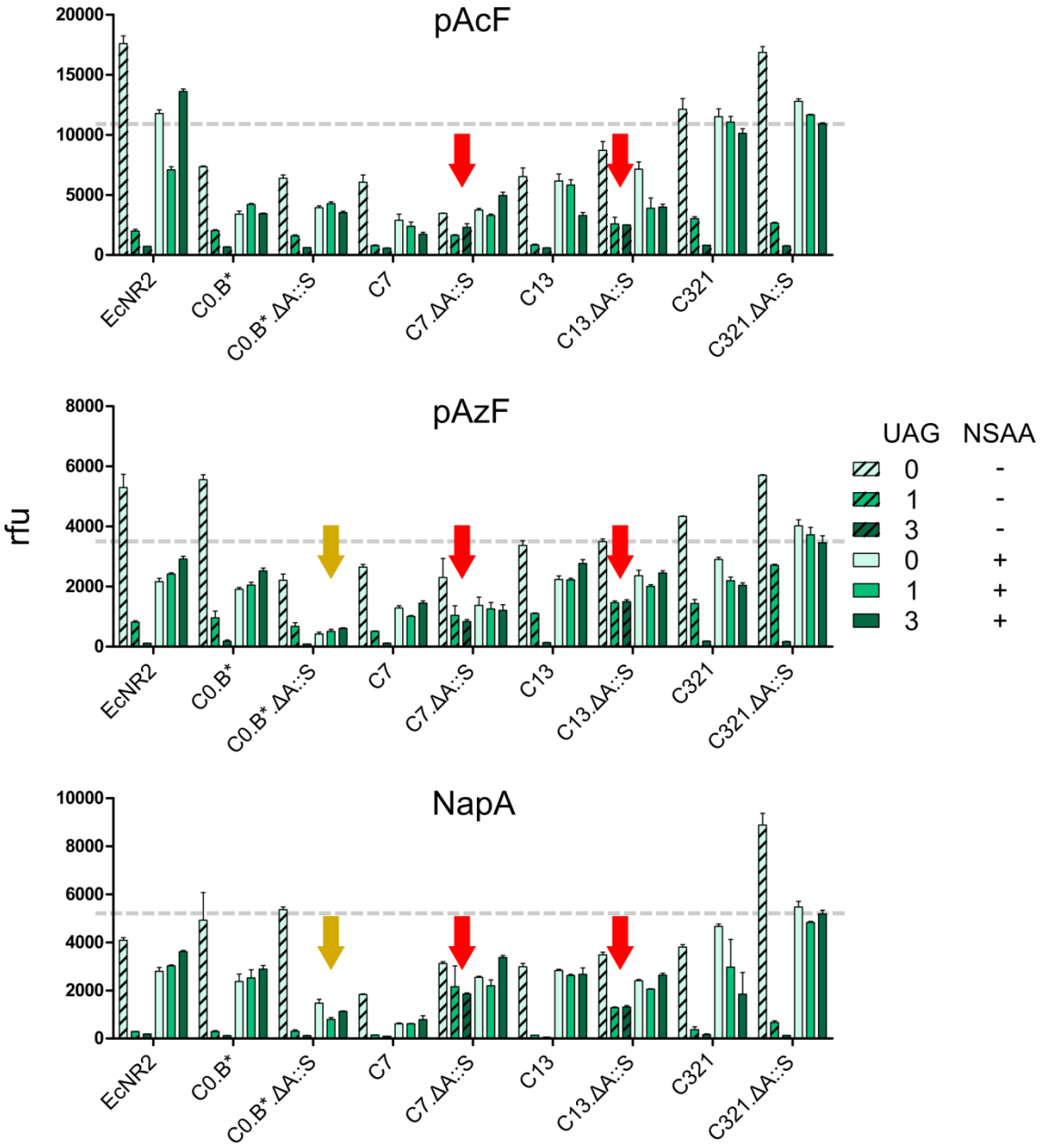


Figure S5-6. Complete removal of all native UAGs permits robust NSAA incorporation. Regardless of whether this is caused by UAG recoding or off-target mutations that non-specifically increase protein production, C321.ΔA::S consistently produces the highest fluorescence after 17 hours of pAcF, pAzF, or NapA expression (see gray dashed horizontal lines as a benchmark). We report raw fluorescence without taking OD600 into account, which may contribute to the reduced fluorescence of the partially recodeded strains. We expressed GFP variants containing 0, 1, or 3 UAG codons in our panel of recoded strains (Table 1) with UAG reassigned to pAcF (top panel; using pEVOL-pAcF (*9*)), pAzF (middle panel; using pEVOL-pCNF), and NapA (bottom panel; using pEVOL-pAcF). As evidenced by strong fluorescence for all reporters in the RF1+ strains, the pEVOL expression system is extremely active and strongly outcompetes RF1 in genes containing up to 3 UAG codons. Notably, C0.B*.ΔA::S yielded less fluorescence than its C0.B* precursor (yellow arrows for pAzF and NapA), probably due to

Figure S5-6 (Continued). toxicity from UAG read-through in essential genes. In contrast, C7.ΔA::S and C13.ΔA::S produced consistent levels of fluorescence in the 1 UAG and 3 UAG GFP reporters even when NSAAs were not supplemented in the media (red arrows), suggesting that these strains have acquired spontaneous UAG suppressors. Unlike the partially recoded strains, C321.ΔA::S yielded robust fluorescence without acquiring a mutational UAG suppressor. Although near-cognate suppression (*18*) may have resulted in residual expression of 1 UAG GFP, the expression was extremely low for 3 UAG GFP.

Spontaneous UAG suppressors in C7.ΔA::S and C13.ΔA::S

GFP fluorescence (Figure S5-6, red arrows) and Western blots (Figure 5-3B) indicated that C7.ΔA::S and C13.ΔA::S had spontaneously acquired efficient natural UAG suppressors. Therefore, we investigated this putative natural suppression in C13.ΔA::S *via* LC-MS/MS. To this end, we expressed an E17* GFP variant in C13.ΔA::S and used LC-MS/MS to identify the amino acid(s) incorporated in response to UAG. This analysis found efficient suppression with Lys, Gln, and Tyr (Table S5-13).

Cells were cultured and lysed as described in the methods section. Cell free extracts were obtained by ultracentrifugation and clarified lysates were applied to Ni-NTA metal affinity resin and purified according to the manufacturer's instructions. Wash buffer contained 50 mM Tris pH 7.5, 500 mM NaCl, 0.5 mM EDTA, 0.5 mM EGTA, 10 mM beta-mercaptoethanol, 50 mM NaF, 1 mM Na_3VO_4 and 5 mM imidazole. Proteins were eluted with buffer containing 500 mM imidazole. Purified protein fractions were subjected to SDS-PAGE electrophoresis, and the gel was stained with Coomassie blue. Protein bands corresponding to the molecular weight of GFP (28.5 kDa) were subjected to in-gel digestion using trypsin as previously described (*61*), and peptides were quantified by UV_{280}. LC-MS was carried out using a 90 min gradient with 100 ng of the digest for each analysis as described above.

Table S5-13. LC-MS/MS of C13.ΔA::S after appearance of natural suppression

Protein	Peptide sequence[a,b]	Exp. MW Da	Calc. MW Da	Δ m ppm	MASCOT Ion score[c]
GFP E17*	**SKGEELFTGVVPILVK**	1714.9869	1714.9869	0.00	38.81
GFP E17*	**GEELFTGVVPILVK**	1499.8608	1499.8599	0.60	29.9
GFP E17*	**SKGEELFTGVVPILVQLDGDVNGHK**	2651.3786	2651.3807	-0.75	84.97
GFP E17*	**SKGEELFTGVVPILVQLDGDVNGHK**	2650.3889	2650.3967	-2.94	68.35
GFP E17*	**GEELFTGVVPILVQLDGDVNGHK**	2435.2598	2435.2697	-4.02	43.59
GFP E17*	**MSKGEELFTGVVPILVQLDGDVNGHK**	2781.4338	2781.4371	-1.19	34.41
GFP E17*	**SKGEELFTGVVPILVYLDGDVNGHK**	2685.3957	2685.4014	-2.12	29.64

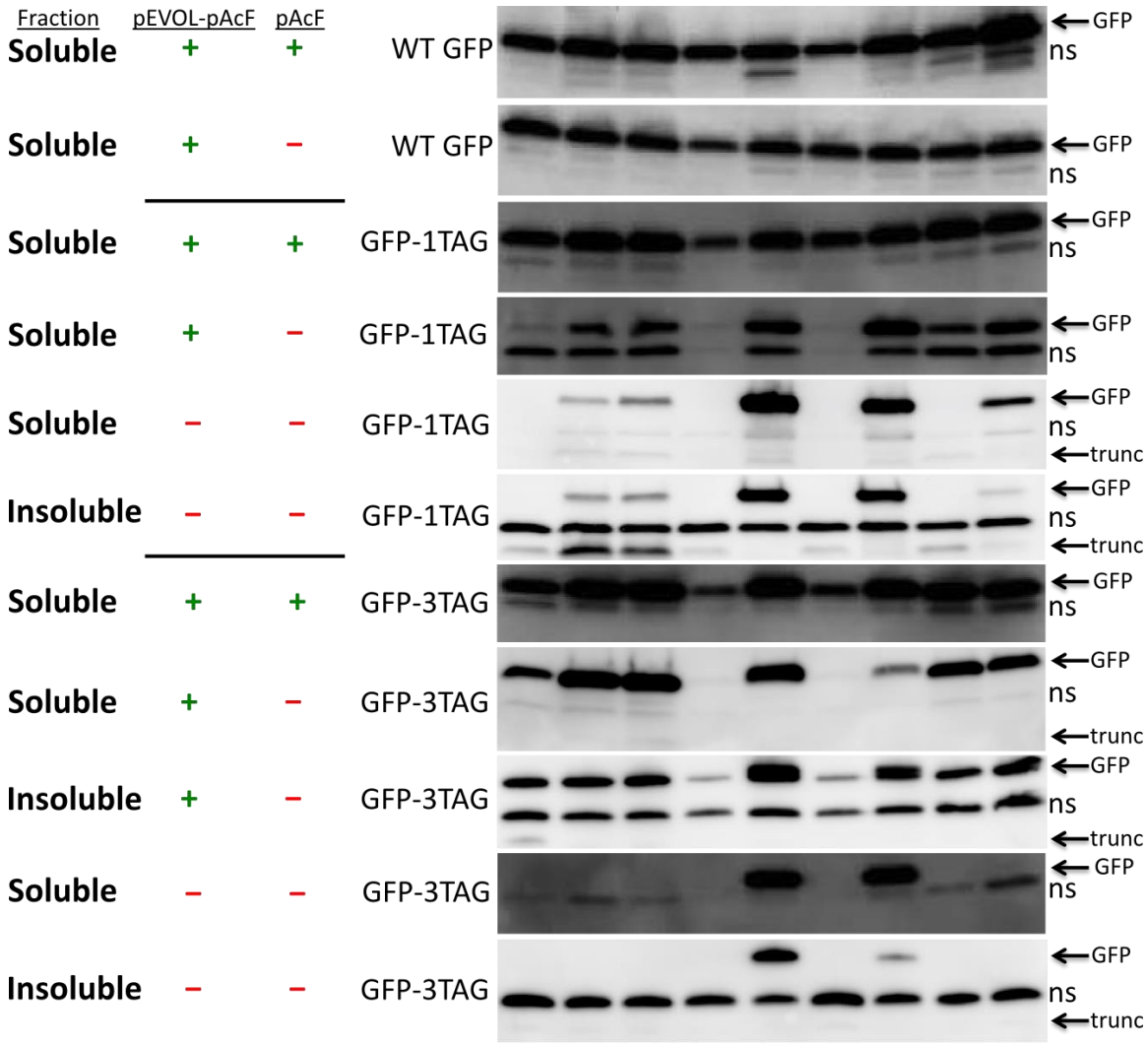
[a]Underlined residues are deamidated.
[b]Lysine (K) insertion adds a unique trypsin cleavage site and produces two unique peptides.
[c]All reported peptides have MASCOT scores above identity.

Western blots: soluble and insoluble fractions

All Western blots not included in Figure 5-3B are included below. Because the anti-GFP antibody binds to an epitope between Y45 and Y151, only the 1UAG GFP variant produced truncation products that could be probed. We tested an anti-His antibody that would recognize the N-terminal 6His tag, but the affinity was too low for robust visualization. The soluble fraction primarily contains full-length GFP, while the insoluble fraction primarily contains the truncation products. Our strain is based on MG1655, which, unlike BL21, does not have important proteases (*lon* and *ompT*) inactivated. Therefore, it is possible that the insoluble truncation products are being degraded and underrepresenting the total amount of UAG-mediated termination.

The supernatant Western blots show that C7.ΔA::S and C13.ΔA::S acquired natural suppressors of UAG, that pEVOL-pAcF is capable of incorporating natural amino acids when pAcF is not supplemented, and that near-cognate UAG suppression (UAG recognition by an anticodon that is not CUA) does not cause strong UAG read-through (Figure S5-18).

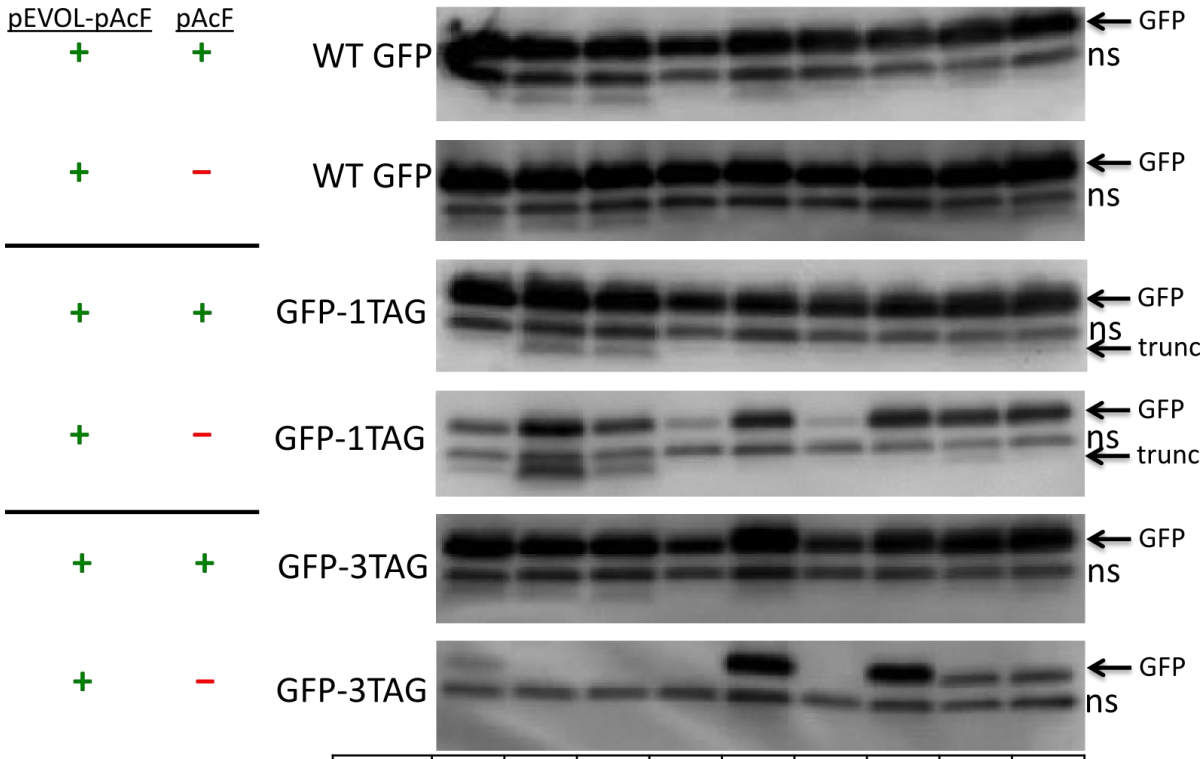
The crude lysate Western blots were performed in an attempt to show the soluble full length GFP and the insoluble truncation products on the same Western blot. Unfortunately, the supernatant overwhelms the insoluble fraction, making it difficult to simultaneous visualization of full-length GFP (soluble) and truncated peptides (insoluble) (Figure S5-19).



UAA	0	0	0	7	7	13	13	321	321
RF2	WT	sup	sup	WT	WT	WT	WT	WT	WT
RF1	WT	WT	S	WT	S	WT	S	WT	S

Figure S5-18. Western blots of GFP variants in the soluble/insoluble fractions. GFP variants containing 0, 1, 2, or 3 UAG codons (Table S5-33) were expressed in recoded strains with UAG reassigned to pAcF (strains harbored pEVOL-pAcF (*9*)). Strain genotypes are indicated as follows: RF1 status is denoted as wt *prfA* (WT), Δ*prfA*::*spec*^{R} (S), Δ*prfA*::*tolC* (T), or Δ*prfA* (Δ). RF2 sup indicates a variant (frameshift removed, T246A, A293E) capable of compensating for RF1 deletion. Western blots of the soluble fraction were probed with an anti-GFP antibody that recognizes an N-terminal epitope. The "ns" signifies a non-specific band. Truncation products ("trunc") were present primarily in the insoluble fractions. Truncation products are most visible for the 1UAG variant because our anti-GFP antibody recognizes an epitope that is not translated in the truncated portion of the 3UAG variant (see Table S5-33 for UAG positions). Still, the 3UAG pellet fractions show faint bands corresponding to the expected size for the 1UAG variant, probably due to read-through at the first UAG. C7.ΔA::S and C13.ΔA::S efficiently produced all variants of GFP regardless of UAG number and pAcF supplementation, suggesting

Figure S5-18 (Continued). that these strains have acquired natural suppressors of UAG. Additionally, full-length 1UAG GFP was visible in all strains lacking RF1 when pEVOL-pAcF was expressed even when pAcF was not supplemented, showing that pEVOL-pAcF is also capable of weakly incorporating natural amino acids. When pEVOL-pAcF was not induced (only expression of constitutive gene copy), a small amount of UAG suppression was observed in C0.B*, C0.B*.ΔA::S, and C321.ΔA::S. This suppression may be caused by weaker expression of the constitutive pAcF-RS copy or by near-cognate suppression (*18*). However, no full-length 3UAG was observed in the absence of pEVOL-pAcF induction and pAcF supplementation, indicating that UAG read-through is weak unless UAG is explicitly reassigned to new function.



UAA	0	0	0	7	7	13	13	321	321
RF2	WT	sup	sup	WT	WT	WT	WT	WT	WT
RF1	WT	WT	S	WT	S	WT	S	WT	S

Figure S5-19. Western blots of GFP variants in a crude lysate. GFP variants containing 0, 1, 2, or 3 UAG codons (Table S5-33) were expressed in recoded strains with UAG reassigned to pAcF (strains harbored pEVOL-pAcF (*9*)). Strain genotypes are indicated as follows: RF1 status is denoted as wt *prfA* (WT), Δ*prfA*::*spec*^{R} (S), Δ*prfA*::*tolC* (T), or Δ*prfA* (Δ). RF2 sup indicates a variant (frameshift removed, T246A, A293E) capable of compensating for RF1 deletion. Western blots of crude lysates were probed with an anti-GFP antibody that recognizes an N-terminal epitope. The "ns" signifies a non-specific band. Truncation products ("trunc") were present in the insoluble fraction, but were faint in the Western blots of crude lysates, perhaps due to proteolysis.

J. Increased T7 resistance

Although T4 bacteriophage did not appear to be affected, T7 bacteriophage exhibited reduced fitness in strains lacking UAG function. Further experimentation is required to fully explain this difference in behavior, but previous work may offer some clues. We have considered which genes might be affected by UAG reassignment for each bacteriophage.

T4: 3 of 19 genes terminating with UAG are essential (Table S5-30a) (*62*).

- Gene 60 (DNA topoisomerase): Gene 60 mRNA contains a short region that must be skipped by translational bypassing in order to produce full length DNA topoisomerase (*63*). A UAG codon plays a role in bypassing efficiency, and UAG stalling may even aid in the translational bypassing.
- Gene 41 (DNA primase/helicase): The C-terminus of gene 41 helicase is involved in Gp59 binding, which is necessary for recombination-dependent replication and for double-strand break repair (*64*). UAG stalling did not significantly impair T4 plaque formation, suggesting that there may have been adequate levels of ribosome rescue by arfA (*65*) and/or yeaJ (*66*) to support normal replication under the conditions tested.
- Gene 15 (Proximal tail sheath stabilizer): Gp15 plays a crucial role in stabilizing the contractile sheath, and forms hexamers that make important contacts with Gp3 and Gp18 (*67*). Hexamer formation occurred even with a C-terminal truncation variant. UAG stalling did not significantly impair T4 plaque formation, suggesting that there may have been adequate levels of ribosome rescue by arfA (*65*) and/or yeaJ (*66*) to support normal tail sheath formation under the conditions tested.

T7: 1 of 6 genes terminating with UAG is essential (Table S5-30b) (*68*).

- Gene 6 (gp6, T7 exonuclease): Gp6 amber mutants are lysis delayed, suggesting that the C-terminus of gp6 may be important for function (*69*). Therefore, ribosome stalling, tmRNA-mediated degradation, and/or C-terminal extension could decrease gp6 activity in the absence of RF1. This in turn could cause a shortage of nucleotides for phage replication and/or inhibit RNA primer removal, recombination, and concatemer processing during T7 replication (*68*).

Table S5-30a. UAG terminating genes in bacteriophage T4 (excerpted from (*62*))

Gene	Essential	Function
60	Yes	DNA topoisomerase subunit
modA.3	No	Hypothetical protein
41	Yes	Replicative and recombination DNA primase/helicase
mobB	No	Putative site-specific intron-like DNA endonuclease
a-gt.2	No	Hypothetical protein
55.8	No	Conserved hypothetical predicted membrane-associated protein
I-TevII	No	Endonuclease for nrdD-intron homing
nrdC.5	No	Conserved hypothetical protein
nrdC.9	No	Conserved hypothetical protein
tk	No	Thymidine kinase
vs.5	No	Conserved hypothetical protein

Table S5-30a (Continued).

e.2	No	Conserved hypothetical predicted membrane-associated protein
5.4	No	Conserved hypothetical protein
15	Yes	Proximal tail sheath stabilizer, connector to gp3 and/or gp19
segD	No	Probable site-specific intron-like DNA endonuclease
uvsY.-2	No	Hypothetical protein
alt.-2	No	Hypothetical protein
I-TevIII	No	Defective intron homing endonuclease
frd.2	No	Conserved hypothetical protein

Table S5-30b. UAG terminating genes in bacteriophage T7 (excerpted from (*68*))

Gene	Essential	Function
0.6B	No	Unknown function
3.8	No	Homing endonuclease
5.3	No	Homing endonuclease
6	Yes	5'->3' dsDNA exonuclease activity, RNase H
18.5	No	Holin (lambda Rz analog)
19.5	No	Holin (suppresses gp17.5 mutants)

Plaque area

RF1$^-$ strains yielded smaller plaques, indicating increased T7 resistance (Figure 5-4). The raw images of plaques on each recoded host are shown in Figure S5-8. We included MG1655 (fastest growth) and C30.5 (slowest growth) as benchmarks to demonstrate that plaque area is not affected by strain doubling time.

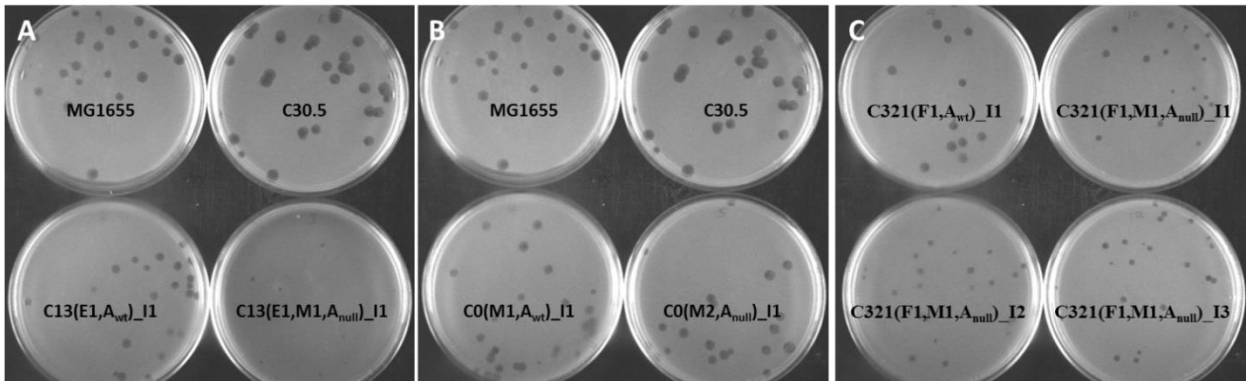


Figure S5-8. Bacteriophage T7 plaques on recoded host strains. With the exception of C0.B*.ΔA::S, all RF1$^-$ strains yielded smaller plaques than their RF1$^+$ counterparts. C13.ΔA::S yielded the smallest plaques, perhaps because translational stalling at native UAG codons may sequester ribosomes and reduce translation or because mutational suppression introduces C-terminal extensions that impair important phage proteins.

Plaque areas were significantly different ($p < 0.0001$) based on RF1 status according to a Kruskal-Wallis one-way ANOVA followed by Dunn's multiple comparison test, where *$p < 0.05$, **$p < 0.01$, and ***$p < 0.001$ (Figure S5-20). The complete results of the multiple comparison test are tabulated in Table S5-14.

Table S5-14. Pairwise statistical comparison of plaque areas.

	C13	C13.ΔA::S	C0.B*	C0.B*.ΔA::S	C321	C321.ΔA::S	C321.ΔA::T	C321.ΔA
MG1655	ns	***	ns	ns	ns	***	***	***
C13		*******	ns	ns	ns	*	*	*
C13.ΔA::S			***	***	***	ns	ns	ns
C0.B*				**ns**	ns	**	**	**
C0.B*.ΔA::S					ns	***	***	***
C321						*******	*******	*******
C321.ΔA::S							ns	ns
C321.ΔA::T								ns
C321.ΔA								

Statistical significances for pair wise plaque area comparisons were calculated using a Kruskal-Wallis one-way ANOVA ($p < 0.0001$) followed by Dunn's multiple comparison test. On the star system, * $p < 0.05$, ** $p < 0.01$, and *** $p < 0.001$. Strains with UAG removed from all essential genes are highlighted in green, strains with a compensatory RF2 variant are highlighted in magenta, and strains with UAG removed from all genes are highlighted in blue. C0.B*.ΔA::S was the only strain that did not show a statistically significant decreased plaque area after RF1 inactivation.

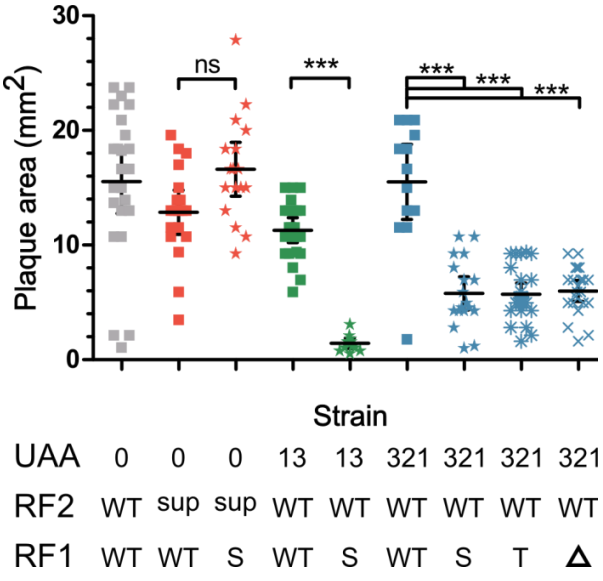


Figure S5-20. Bacteriophage T7 infection is attenuated in GROs lacking RF1. RF1 (*prfA*) status is denoted by symbol shape: ■ is wt *prfA* (WT), ★ is Δ*prfA*::*spec*^{R} (S), ❋ is Δ*prfA*::*tolC* (T), and ✖ is a clean deletion of *prfA* (Δ). RF2 "sup" indicates a variant (frameshift removed, T246A, A293E) capable of suppressing lethality of RF1 deletion. **(A)** Plaque area (mm^2) distributions for strains with or without RF1. Plaque areas were calculated using ImageJ, and means +/- 95% confidence intervals are presented with the raw plaque area data ($n > 12$ for each strain). In the absence of RF1, all strains except for C0.B*.ΔA::S yielded significantly smaller plaques, indicating that the RF2 variant can terminate UAG adequately to maintain T7 fitness. Statistics are based on a Kruskal-Wallis one-way ANOVA followed by Dunn's multiple comparison test (*$p < 0.05$, **$p < 0.01$, and ***$p < 0.001$). A statistical summary can be found in Table S5-8.

Kinetic lysis

To confirm the plaque area observations, we performed kinetic lysis curves with T7 infected at a multiplicity of infection (MOI) of 5. This ensured that all host cells were rapidly and synchronously infected by phage particles. We monitored lysis on a Biotek H4 plate reader with OD_{600} measurements taken every 5 minutes. The mean lysis curves were plotted using average OD_{600} values for each time point (n = 12), and a two-way ANOVA showed that the lysis curves were significantly different ($p < 0.0001$). Each lysis curve was fit to a cumulative normal distribution from which mean lysis parameters were calculated using the normcdf function in MATLAB (Figure S5-9).

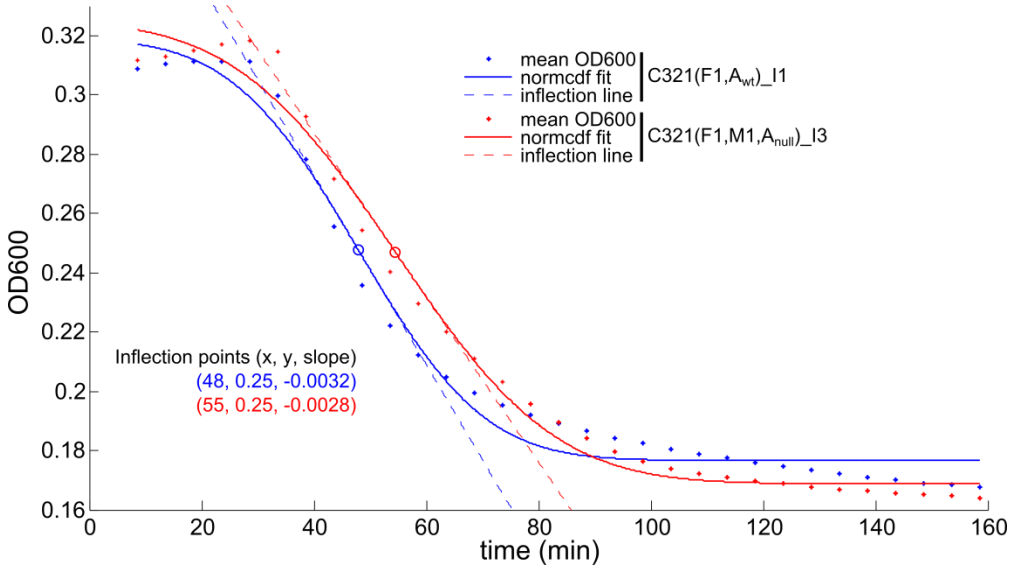


Figure S5-9. T7 kinetic lysis curves (MOI = 5). Mean lysis time (+/- standard error of the mean) was 47.9 (+/- 0.1) minutes for C321 and 54.5 (+/- 0.2) minutes for C321.ΔA::S, indicating that lysis is delayed in the absence of RF1 (n = 12, $p < 0.0001$, unpaired t test with Welch's correction). Mean lysis OD_{600} was 0.25 (with negligible standard error of the mean) for both strains, showing that both hosts were infected under identical conditions and could be completely lysed by T7.

One-step growth curves

To determine burst size and eclipse time, one step growth curves were performed as previously described (*70*). Briefly, mid-log phase cultures were infected at MOI = 0.1. At 3 minutes post infection, 30 µL of infected culture was diluted 500-fold into 15 mL LB^L to minimize further phage adsorption. Two aliquots were taken at t = 6 minutes—one aliquot was titered directly and the other was treated with chloroform before titering. Adsorption efficiency was determined by $(pfu_{noCHCL3} - pfu_{CHCl3})/ pfu_{noCHCL3}$. Additional aliquots of the infection were taken at the following time points and were immediately treated with chloroform to release intracellular phage particles and to halt infection: 6, 18, 21, 24, 27, 29, 31, 33, 35, 37, 39, 41, 45, and 50 minutes. These samples were then titered to monitor intracellular phage assembly during a single phage life cycle. Six replicates were performed, and each one-step growth curve was analyzed separately before their parameters were averaged. We estimated one-step growth parameters by using the scipy.optimize.curve_fit function to fit pfu/mL to

$$\phi = \begin{cases} 0, & 0 < t < a \\ r(t-a), & a < t < \frac{B}{r} + a \\ B, & t > \frac{B}{r} + a \end{cases}$$

where ϕ is the number of phage progeny as a function of time (t), a is eclipse time, r is rise rate, and B is burst size (*70*).

Adsorption efficiency ranged from ~20% – 60%, which is considerably lower than the >95% that we observed during the T7 fitness assay (Figure 5-4C). This discrepancy is probably because we performed this assay using phage lysate that had been stored at 4 °C for several weeks. Although we re-titered before each replicate, infection became less efficient as the phage lysate aged. Although this increased variance, T7 infection consistently proceeded more efficiently in C321 than in C321.ΔA (Figure S5-21).

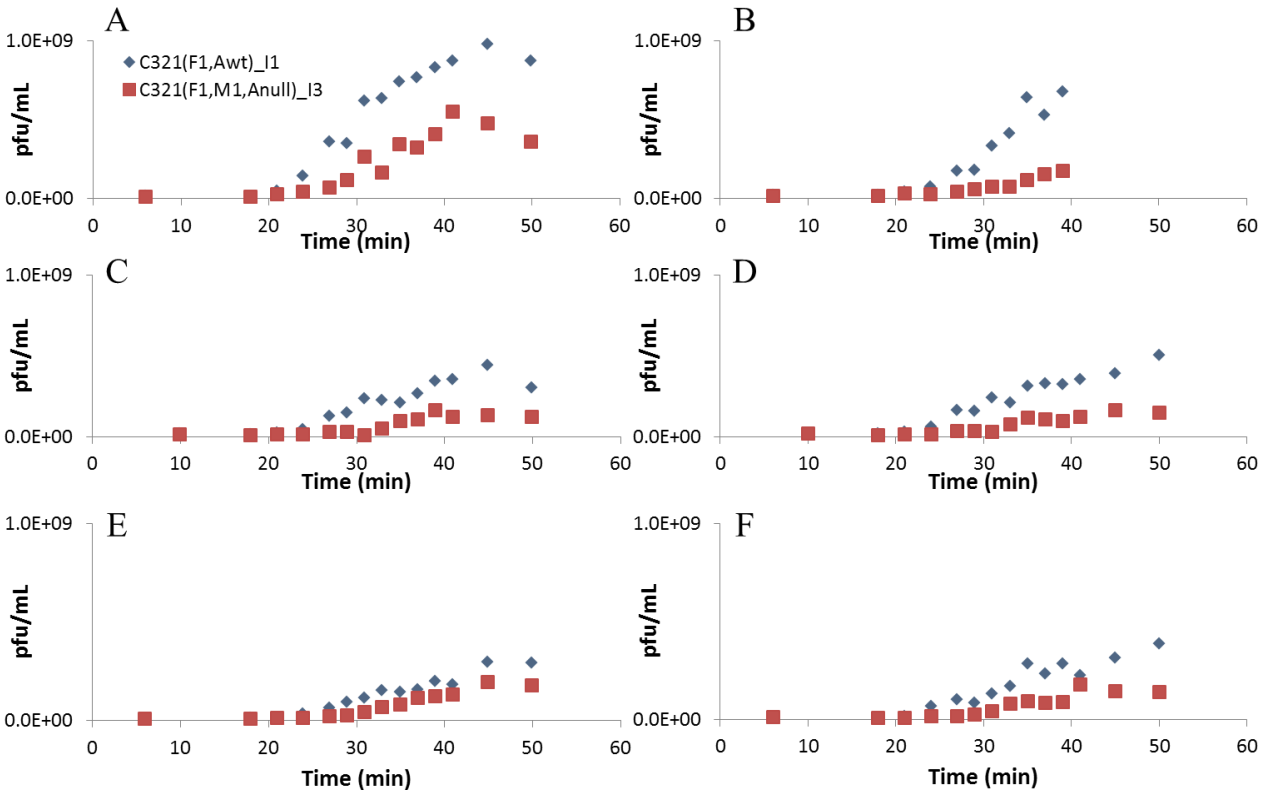


Figure S5-21. One step growth curves were performed using hosts C321 and C321.ΔA in six replicates: A = replicate 1, B = replicate 2, C = replicate 3, D = replicate 4, E = replicate 5, and F = replicate 6. Although the T7 lysate was properly stored at 4 °C in LB^{L} supplemented with 900 mM sodium chloride, we found that longterm storage decreased adsorption efficiency (and apparent burst size) in both hosts. Therefore, replicates 3 and 4 were taken on the same day and replicates 5 and 6 were taken two days later to minimize variance. Even despite the variance, C321 consistently yielded larger burst sizes than C321.ΔA in all replicates.

Because the first two replicates yielded higher phage titers than the others, we combined replicates 3-6, which were performed over the course of four days (Figure S5-10). RF1 removal caused a 30% longer eclipse time (p = 0.01), a 60% smaller burst size (p = 0.02), and a 35% slower rise rate (p = 0.04) (Figure S5-22, Table S5-15). Percentage changes were calculated by $(RF1^{-}_{param} - RF1^{+}_{param})/ RF1^{+}_{param}$.

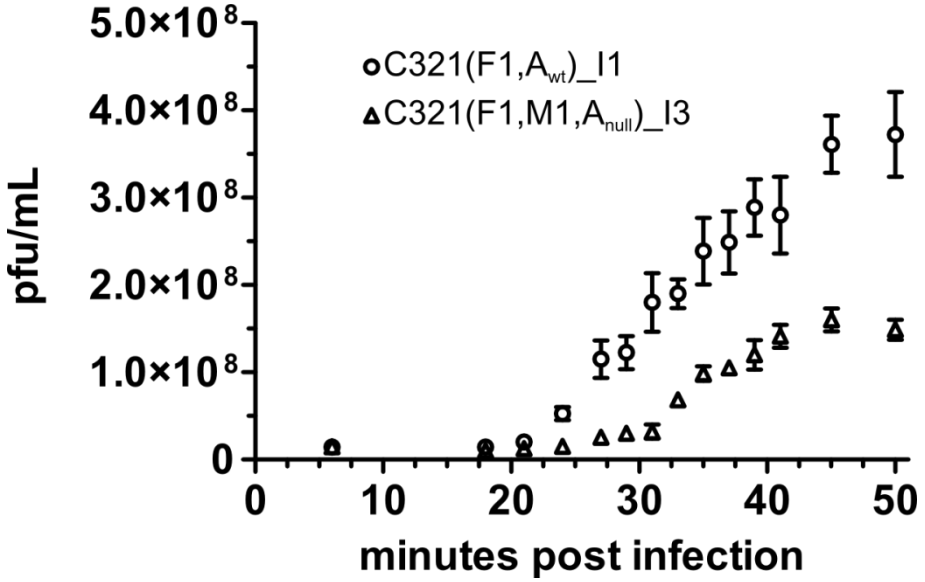


Figure S5-10. One step growth curve averaged across replicates 3-6 (Figure S5-21C-F). Mean pfu/mL +/- SEM are plotted for each time point. The one step growth curves for each host were significantly different (p = 0.002), as determined by a two way repeated measures ANOVA.

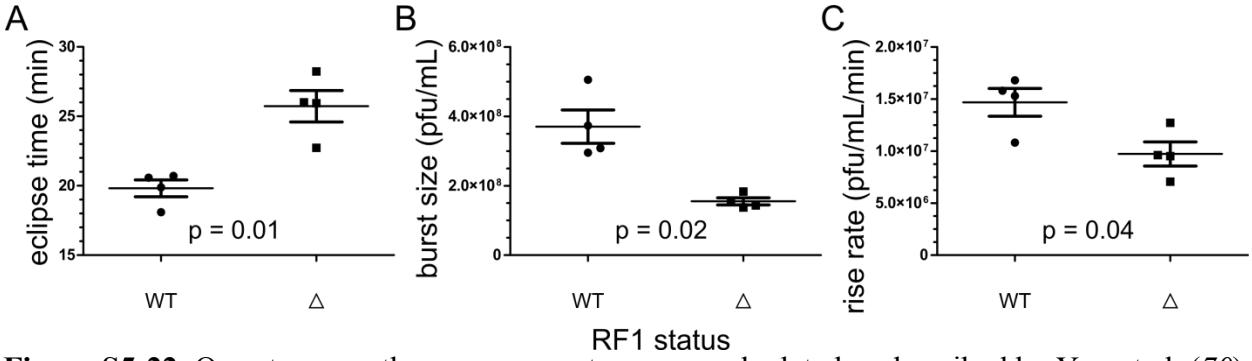


Figure S5-22. One step growth curve parameters were calculated as described by You et al. (*70*). Raw data points are plotted with mean +/- SEM. The p values were calculated using an unpaired t test with Welch's correction. Compared to C321 (WT), the C321.ΔA (Δ) supports T7 infection with (A) a 30% (+/- 2%) longer eclipse time (p = 0.01), (B) a 59% (+/- 9%) smaller burst size (p = 0.02), and (C) a 35% (+/- 5%) slower rise rate (p = 0.04).

Table S5-15. One-step growth parameters: eclipse time, rise rate, and burst size

Metric[a]	**C321**	**C321.ΔA**	**% change**[b]	**p value**[c]
Eclipse time (min)	19.8 (+/- 0.6)	25.7 (+/- 1.1)	30% longer	0.01
Burst size (pfu/mL)	3.7E8 (+/- 4.8E7)	1.5E8 (+/- 1.0E7)	59% smaller	0.02
Rise rate (pfu/mL/min)	1.5E7 (+/- 1.3E6)	9.7E6 (+/- 1.2E6)	35% slower	0.04

[a] Data is based on 4 replicates (Replicates 3-6, Figure S5-21C-F)
[b] % change in C321.ΔA with respect to C321; $(RF1^{-}_{param} - RF1^{+}_{param})/ RF1^{+}_{param}$
[c] p values were calculated using an unpaired t test with Welch's correction

K. Selectable markers used in this study

>*mutS*::*cat* (1017 bp)
gctgatccactagttctagagcggccgctctaaaaaatgccctcttgggttatcaagagggtcattatatttcg
cggaattcatgctatcgacgtcgatatctggcgaaaatgagacgttgatcggcacgtaagaggttccaactttc
accataatgaaataagatcactaccgggcgtatttttgagttatcgagattttcaggagctaaggaagctaaa
atggagaaaaaaatcactggatataccaccgttgatatatcccaatggcatcgtaaagaacattttgaggcatt
tcagtcagttgctcaatgtacctataaccagaccgttcagctggatattacggccttttaaagaccgtaaaga
aaaataagcacaagttttatccggcctttattcacattcttgcccgcctgatgaatgctcatccggaatttcgt
atggcaatgaaagacggtgagctggtgatatgggatagtgttcacccttgttacaccgttttccatgagcaaac
tgaaacgttttcatcgctctggagtgaataccacgacgatttccggcagtttctacacatatattcgcaagatg
tggcgtgttacggtgaaaacctggcctatttccctaaagggtttattgagaatatgtttttcgtctcagccaat
ccctgggtgagtttcaccagttttgatttaaacgtggccaatatggacaacttcttcgcccccgttttcaccat
gggcaaatattatacgcaaggcgacaaggtgctgatgccgctggcgattcaggttcatcatgccgtctgtgatg
gcttccatgtcggcagaatgcttaatgaattacaacagtactgcgatgagtggcagggcggggcgtaatttgat
atcgagctcgtcagcaggcgcgcctgtaatcacactggctcaccttcgggtgggcctttctgcgtttaaaaaaa
acgggccggcgcgaacgccggcccgcggccgccacccagcttttgttccctttag

>*kan*^{R}-oriT (1949 bp); oriT is from RK2 (*28*)
cctgtgacggaagatcacttcgcagaataaataaatcctggtgtccctgttgataccgggaagccctgggccaa
cttttggcgaaaatgagacgttgatcggcacgtaagaggttccaactttcaccataatgaaataagatcactac
cgggcgtatttttgagttgtcgagattttcaggagctaaggaagctaaaatgagccatattcaacgggaaacg
tcgaggccgcgattaaattccaacatggatgctgatttatatgggtataaatgggctcgcgataatgtcgggca
atcaggtgcgacaatctatcgcttgtatgggaagcccgatgcgccagagttgtttctgaaacatggcaaaggta
gcgttgccaatgatgttacagatgagatggtcagactaaactggctgacggaatttatgcctcttccgaccatc
aagcattttatccgtactcctgatgatgcatggttactcaccactgcgatccccggaaaaacagcattccaggt
attagaagaatatcctgattcaggtgaaaatattgttgatgcgctggcagtgttcctgcgccggttgcattcga
ttcctgtttgtaattgtccttttaacagcgatcgcgtatttcgtctcgctcaggcgcaatcacgaatgaataac
ggtttggttgatgcgagtgattttgatgacgagcgtaatggctggcctgttgaacaagtctggaaagaaatgca
taaacttttgccattctcaccggattcagtcgtcactcatggtgatttctcacttgataaccttatttttgacg
aggggaaattaataggttgtattgatgttggacgagtcggaatcgcagaccgataccaggatcttgccatccta
tggaactgcctcggtgagttttctccttcattacagaaacggctttttcaaaaatatggtattgataatcctga
tatgaataaattgcagtttcatttgatgctcgatgagtttttctaatttttttaaggcagttattggtgccctt
aaacgcctggttgctacgcctgaataagtgataataagcggatgaatggcagaaattcgaaagcaaattcgacc
cggtcgtcggttcagggcagggtcgttaaatagccgcttatgtctattgctggttggcgctcggtcttgccttg
ctcgtcggtgatgtacttcaccagctccgcgaagtcgctcttcttgatggagcgcatggggacgtgcttggcaa
tcacgcgcacccccggccgttttagcggctaaaaaagtcatggctctgccctcgggcggaccacgcccatcat
gaccttgccaagctcgtcctgcttctcttcgatcttcgccagcagggcgaggatcgtggcatcaccgaaccgcg
ccgtgcgcgggtcgtcggtgagccagagtttcagcaggccgcccaggcggcccaggtcgccattgatgcgggcc
agctcgcggacgtgctcatagtccacgacgcccgtgattttgtagccctggccgacggccagcaggtaggccga
caggctcatgccggccgccgccgccttttcctcaatcgctcttcgttcgtctggaaggcagtacaccttgatag
gtgggctgcccttcctggttggcttggtttcatcagccatccgcttgccctcatctgttacgccggcggtagcc
ggccagcctcgcagagcaggattcccgttgagcaccgccaggtgcgaataagggacagtgaagaaggaacaccc
gctcgcgggtgggcctacttcacctatcctgcccggctgacgccgttggatacaccaaggaaagtctacacgaa
ccctttggcaaaatcctgtatatcgtgcgaaaaaggatggatataccgaaaaaatcgctataatgaccccgaag
cagggttatgcagcggaaaagcgct

>*gent*^{R} (831 bp)
acgcacaccgtggaaacggatgaaggcacgaacccagttgacataagcctgttcggttcgtaaactgtaatgca
agtagcgtatgcgctcacgcaactggtccagaaccttgaccgaacgcagcggtggtaacggcgcagtggcggtt
ttcatggcttgttatgactgtttttttgtacagtctatgcctcgggcatccaagcagcaagcgcgttacgccgt
gggtcgatgtttgatgttatggagcagcaacgatgttacgcagcagcaacgatgttacgcagcagggcagtcgc
cctaaaacaaagttaggtggctcaagtatgggcatcattcgcacatgtaggctcggccctgaccaagtcaaatc
catgcgggctgctcttgatcttttcggtcgtgagttcggagacgtagccacctactcccaacatcagccggact

ccgattacctcgggaacttgctccgtagtaagacattcatcgcgcttgctgccttcgaccaagaagcggttgtt
ggcgctctcgcggcttacgttctgcccaggtttgagcagccgcgtagtgagatctatatctatgatctcgcagt
ctccggcgagcaccggaggcagggcattgccaccgcgctcatcaatctcctcaagcatgaggccaacgcgcttg
gtgcttatgtgatctacgtgcaagcagattacggtgacgatcccgcagtggctctctatacaaagttgggcata
cgggaagaagtgatgcactttgatatcgacccaagtaccgccacctaacaattcgttcaagccgagatcggctt
cccgg

>*zeo*^{R} (761bp)
ggtgttgacaattaatcatcggcatagtatatcggcatagtataatacgacaaggtgaggaactaaaccatggc
caagttgaccagtgccgttccggtgctcaccgcgcgcgacgtcgccggagcggtcgagttctggaccgaccggc
tcgggttctcccgggacttcgtggaggacgacttcgccggtgtggtccgggacgacgtgaccctgttcatcagc
gcggtccaggaccaggtggtgccggacaacaccctggcctgggtgtgggtgcgcggcctggacgagctgtacgc
cgagtggtcggaggtcgtgtccacgaacttccgggacgcctccgggccggccatgaccgagatcggcgagcagc
cgtgggggcgggagttcgccctgcgcgacccggccggcaactgcgtgcacttcgtggccgaggagcaggactga
cacgtccgacggcggcccacgggtcccaggcctcggagatccgtccccctttccctttgtcgatatcatgtaat
tagttatgtcacgcttacattcacgccctccccccacatccgctctaaccgaaaaggaaggagttagacaacct
gaagtctaggtccctatttatttttttatagttatgttagtattaagaacgttatttatatttcaaatttttct
tttttttctgtacagacgcgtgtacgcatgtaacattatactgaaaaccttgcttgagaaggttttgggacgct
cgaaggctttaatttgcaagct

>*spec*^{R} (1201 bp)
cagccaggacagaaatgcctcgacttcgctgctgcccaaggttgccgggtgacgcacaccgtggaaacggatga
aggcacgaacccagtggacataagcctgttcggttcgtaagctgtaatgcaagtagcgtatgcgctcacgcaac
tggtccagaaccttgaccgaacgcagcggtggtaacggcgcagtggcggttttcatggcttgttatgactgttt
ttttggggtacagtctatgcctcgggcatccaagcagcaagcgcgttacgccgtgggtcgatgtttgatgttat
ggagcagcaacgatgttacgcagcagggcagtcgccctaaaacaaagttaaacatcatgagggaagcggtgatc
gccgaagtatcgactcaactatcagaggtagttggcgtcatcgagcgccatctcgaaccgacgttgctggccgt
acatttgtacggctccgcagtggatggcggcctgaagccacacagtgatattgatttgctggttacggtgaccg
taaggcttgatgaaacaacgcggcgagctttgatcaacgacctttttggaaacttcggcttcccctggagagagc
gagattctccgcgctgtagaagtcaccattgttgtgcacgacgacatcattccgtggcgttatccagctaagcg
cgaactgcaatttggagaatggcagcgcaatgacattcttgcaggtatcttcgagccagccacgatcgacattg
atctggctatcttgctgacaaaagcaagagaacatagcgttgccttggtaggtccagcggcggaggaactcttt
gatccggttcctgaacaggatctatttgaggcgctaaatgaaaccttaacgctatggaactcgccgcccgactg
ggctggcgatgagcgaaatgtagtgcttacgttgtcccgcatttggtacagcgcagtaaccggcaaaatcgcgc
cgaaggatgtcgctgccgactgggcaatggagcgcctgccggcccagtatcagcccgtcatacttgaagctaga
caggcttatcttggacaagaagaagatcgcttggcctcgcgcgcagatcagttggaagaatttgtccactacgt
gaaaggcgagatcaccaaggtagtcggcaaataaagctttactgagctaataacaggactgctggtaatcgcag
gccttttttatttctgca

>*tolC* (1764 bp)
ttgaggcacattaacgccctatggcacgtaacgccaacctttgcggtagcggcttctgctagaatccgcaata
attttacagtttgatcgcgctaaatactgcttcaccacaaggaatgcaaatgaagaaattgctccccattctta
tcggcctgagcctttctgggttcagttcgttgagccaggccgagaacctgatgcaagtttatcagcaagcacgc
cttagtaacccggaattgcgtaagtctgccgccgatcgtgatgctgcctttgaaaaaattaatgaagcgcgcag
tccattactgccacagctaggtttaggtgcagattacacctatagcaacggctaccgcgacgcgaacggcatca
actctaacgcgaccagtgcgtccttgcagttaactcaatccatttttgatatgtcgaaatggcgtgcgttaacg
ctgcaggaaaaagcagcagggattcaggacgtcacgtatcagaccgatcagcaaaccttgatcctcaacaccgc
gaccgcttatttcaacgtgttgaatgctattgacgttctttcctatacacaggcacaaaaagaagcgatctacc
gtcaattagatcaaaccacccaacgttttaacgtgggcctggtagcgatcaccgacgtgcagaacgcccgcgca
cagtacgataccgtgctggcgaacgaagtgaccgcacgtaataaccttgataacgcggtagagcagctgcgcca
gatcaccggtaactactatccggaactggctgcgctgaatgtcgaaaactttaaaaccgacaaaccacagccgg
ttaacgcgctgctgaaagaagccgaaaaacgcaacctgtcgctgttacaggcacgcttgagccaggacctggcg
cgcgagcaaattcgccaggcgcaggatggtcacttaccgactctggatttaacggcttctaccgggatttctga
cacctcttatagcggttcgaaaacccgtggtgccgctggtacccagtatgacgatagcaatatgggccagaaca

aagttggcctgagcttctcgctgccgatttatcagggcggaatggttaactcgcaggtgaaacaggcacagtac
aactttgtcggtgccagcgagcaactggaaagtgcccatcgtagcgtcgtgcagaccgtgcgttcctccttcaa
caacattaatgcatctatcagtagcattaacgcctacaaacaagccgtagtttccgctcaaagctcattagacg
cgatggaagcgggctactcggtcggtacgcgtaccattgttgatgtgttggatgcgaccaccacgttgtacaac
gccaagcaagagctggcgaatgcgcgttataactacctgattaatcagctgaatattaagtcagctctgggtac
gttgaacgagcaggatctgctggcactgaacaatgcgctgagcaaaccggtttccactaatccggaaaacgttg
caccgcaaacgccggaacagaatgctattgctgatggttatgcgcctgatagcccggcaccagtcgttcagcaa
acatccgcacgcactaccaccagtaacggtcataacccttccgtaactgaaagcttgatatcgaattcctgca
gcccgggggatcccatggtacgcgtgctagaggcatcaaataaaacgaaaggctcagtcgaaagactgggcctt
tcgttttatctgttgtttgtcggtgaacgctctcctgagtaggacaaatccgccgccctaga

>*galK* (1270 bp)

cctgttgacaattaatcatcggcatagtatatcggcatagtataatacgacaaggtgaggaactaaacccagga
ggcagatcatgagtctgaaagaaaaaacacaatctctgtttgccaacgcatttggctaccctgccactcacacc
attcaggcgcctggccgcgtgaatttgattggtgaacacaccgactacaacgacggtttcgttctgccctgcgc
gattgattatcaaaccgtgatcagttgtgcaccacgcgatgaccgtaaagttcgcgtgatggcagccgattatg
aaaatcagctcgacgagttttccctcgatgcgcccattgtcgcacatgaaaactatcaatgggctaactacgtt
cgtggcgtggtgaaacatctgcaactgcgtaacaacagcttcggcggcgtggacatggtgatcagcggcaatgt
gccgcagggtgccgggttaagttcttccgcttcactggaagtcgcggtcggaaccgtattgcagcagctttatc
atctgccgctggacggcgcacaaatcgcgcttaacggtcaggaagcagaaaaccagtttgtaggctgtaactgc
gggatcatggatcagctaatttccgcgctcggcaagaaagatcatgccttgctgatcgattgccgctcactggg
gaccaaagcagtttccatgcccaaaggtgtggctgtcgtcatcatcaacagtaacttcaaacgtaccctggttg
gcagcgaatacaacacccgtcgtgaacagtgcgaaaccggtgcgcgtttcttccagcagccagccctgcgtgat
gtcaccattgaagagttcaacgctgttgcgcatgaactggacccgatcgtggcaaaacgcgtgcgtcatatact
gactgaaaacgcccgcaccgttgaagctgccagcgcgctggagcaaggcgacctgaaacgtatgggcgagttga
tggcggagtctcatgcctctatgcgcgatgatttcgaaatcaccgtgccgcaaattgacactctggtagaaatc
gtcaaagctgtgattggcgacaaaggtggcgtacgcatgaccggcggcggatttggcggctgtatcgtcgcgct
gatcccggaagagctggtgcctgccgtacagcaagctgtcgctgaacaatatgaagcaaaaacaggtattaaag
agacttttacgtttgtaaaccatcacaaggagcaggacagtgctgaaaaaaaaaaccccgcccctgacagggc
ggggttttttt

>*malK* (1116 bp)

atggcgagcgtacagctgcaaaatgtaacgaaagcctggggcgaggtcgtggtatcgaaagatatcaatctcga
tatccatgaaggtgaattcgtggtgtttgtcggaccgtctggctgcggtaaatcgactttactgcgcatgattg
ccgggcttgagacgatcaccagcggcgacctgttcatcggtgagaaacggatgaatgacactccgccagcagaa
cgcggcgttggtatggtgtttcagtcttacgcgctctatccccacctgtcagtagcagaaaacatgtcatttgg
cctgaaactggctggcgcaaaaaaagaggtgattaaccaacgcgttaaccaggtggcggaagtgctacaactgg
cgcatttgctggatcgcaaaccgaaagcgctctccggtggtcagcgtcagcgtgtggcgattggccgtacgctg
gtggccgagccaagcgtatttttgctcgatgaaccgctctccaacctcgatgctgcactgcgtgtgcaaatgcg
tatcgaaatctcccgtctgcataaacgcctgggccgcacaatgatttacgtcacccacgatcaggtcgaagcga
tgacgctggccgacaaaatcgtggtgctggacgccggtcgcgtggcgcaggttgggaaaccgctggagctgtac
cactatccggcagaccgttttgtcgccggatttatcggttcgccaaagatgaacttcctgccggtaaaagtgac
cgccaccgcaatcgatcaagtgcaggtggagctgccgatgccaaatcgtcagcaagtctggctgccagttgaaa
gccgtgatgtccaggttggagccaatatgtcgctgggtattcgcccggaacatctactgccgagtgatatcgct
gacgtcatccttgagggtgaagttcaggtcgtcgagcaactcggcaacgaaactcaaatccatatccagatccc
ttccattcgtcaaaacctggtgtaccgccagaacgacgtggtgttggtagaagaaggtgccacattcgctatcg
gcctgccgccagagcgttgccatctgttccgtgaggatggcactgcatgtcgtcgactgcataaggagccgggc
gtttaa

Table S5-33. Sequences of GFP variants containing UAG codons (UAG codons are highlighted in red).

>GFP-NHis-0UAG
atgcaccaccaccaccaccacagtaaaggagaagaacttttcactggagttgtcccaattcttgttgaattagatggtgatgttaatgggcaca
aattttctgtcagtggagagggtgaaggtgatgcaacatacggaaaacttacccttaaatttatttgcactactggaaaactacctgttccatgg
ccaacacttgtcactactttctcttatggtgttcaatgcttttcccgttatccggatcacatgaaacggcatgacttttcaagagtgccatgcccg
aaggttatgtacaggaacgcactatatctttcaaagatgacgggaactacaagacgcgtgctgaagtcaagtttgaaggtgataccttgtta
atcgtatcgagttaaaaggtattgattttaaagaagatggaaacattctcggacacaaactcgaatacaactataactcacacaatgtatacatc
acggcagacaaacaaaagaatggaatcaaagctaacttcaaaattcgccacaacattgaagatggatccgttcaactagcagaccattatca
acaaaatactccaattggcgatggccctgtccttttaccagacaaccattacctgtcgacacaatctgccctttcgaaagatcccaacgaaaa
gcgtgaccacatggtccttcttgagtttgtaactgctgctgggattacacatggcatggatgagctctacaaataa

>GFP-NHis-1UAG
atgcaccaccaccaccaccacagtaaaggagaagaacttttcactggagttgtcccaattcttgttgaattagatggtgatgttaatgggcaca
aattttctgtcagtggagagggtgaaggtgatgcaacatacggaaaacttacccttaaatttatttgcactactggaaaactacctgttccatgg
ccaacacttgtcactactttctcttatggtgttcaatgcttttcccgttatccggatcacatgaaacggcatgacttttcaagagtgccatgcccg
aaggttatgtacaggaacgcactatatctttcaaagatgacgggaacftacaagacgcgtgctgaagtcaagtttgaaggtgataccttgtta
atcgtatcgagttaaaaggtattgattttaaagaagatggaaacattctcggacacaaactcgaatacaactataactcacacaatgtatagatc
acggcagacaaacaaaagaatggaatcaaagctaacttcaaaattcgccacaacattgaagatggatccgttcaactagcagaccattatca
acaaaatactccaattggcgatggccctgtccttttaccagacaaccattacctgtcgacacaatctgccctttcgaaagatcccaacgaaaa
gcgtgaccacatggtccttcttgagtttgtaactgctgctgggattacacatggcatggatgagctctacaaataa

>GFP-NHis-2UAG
atgcaccaccaccaccaccacagtaaaggagaagaacttttcactggagttgtcccaattcttgttgaattagatggtgatgttaatgggcaca
aattttctgtcagtggagagggtgaaggtgatgcaacatagggaaaacttacccttaaatttatttgcactactggaaaactacctgttccatgg
ccaacacttgtcactactttctcttatggtgttcaatgcttttcccgttatccggatcacatgaaacggcatgacttttcaagagtgccatgcccg
aaggttatgtacaggaacgcactatatctttcaaagatgacgggaactacaagacgcgtgctgaagtcaagtttgaaggtgataccttgtta
atcgtatcgagttaaaaggtattgattttaaagaagatggaaacattctcggacacaaactcgaatacaactataactcacacaatgtatacatc
acggcagacaaacaaaagaatggaatcaaagctaacttcaaaattcgccacaacattgaagatggatccgttcaactagcagaccattagc
aacaaaatactccaattggcgatggccctgtccttttaccagacaaccattacctgtcgacacaatctgccctttcgaaagatcccaacgaaaa
gcgtgaccacatggtccttcttgagtttgtaactgctgctgggattacacatggcatggatgagctctacaaataa

>GFP-NHis-3UAG
atgcaccaccaccaccaccacagtaaaggagaagaacttttcactggagttgtcccaattcttgttgaattagatggtgatgttaatgggcaca
aattttctgtcagtggagagggtgaaggtgatgcaacatagggaaaacttacccttaaatttatttgcactactggaaaactacctgttccatgg
ccaacacttgtcactactttctcttatggtgttcaatgcttttcccgttatccggatcacatgaaacggcatgacttttcaagagtgccatgcccg
aaggttatgtacaggaacgcactatatctttcaaagatgacgggaactacaagacgcgtgctgaagtcaagtttgaaggtgataccttgtta
atcgtatcgagttaaaaggtattgattttaaagaagatggaaacattctcggacacaaactcgaatacaactataactcacacaatgtatagatc
acggcagacaaacaaaagaatggaatcaaagctaacttcaaaattcgccacaacattgaagatggatccgttcaactagcagaccattagc
aacaaaatactccaattggcgatggccctgtccttttaccagacaaccattacctgtcgacacaatctgccctttcgaaagatcccaacgaaaa
gcgtgaccacatggtccttcttgagtttgtaactgctgctgggattacacatggcatggatgagctctacaaataa

L. References

1. K. Vetsigian, C. Woese, N. Goldenfeld, Collective evolution and the genetic code. *PNAS* **103**, 10696 (2006).
2. D. V. Goeddel, D. G. Kleid, F. Bolivar, H. L. Heyneker, D. G. Yansura, R. Crea, T. Hirose, A. Kraszewski, K. Itakura, A. D. Riggs, Expression in Escherichia coli of Chemically Synthesized Genes for Human Insulin. *PNAS* **76**, 106 (1979).
3. D. C. Krakauer, V. A. A. Jansen, Red queen dynamics of protein translation. *J. Theor. Biol.* **218**, 97 (2002).
4. M. G. Schafer, A. A. Ross, J. P. Londo, C. A. Burdick, E. H. Lee, S. E. Travers, P. K. Van de Water, C. L. Sagers, The Establishment of Genetically Engineered Canola Populations in the U.S. *PLoS One* **6**, e25736 (2011).
5. J. M. Sturino, T. R. Klaenhammer, Engineered bacteriophage-defence systems in bioprocessing. *Nat. Rev. Microbiol.* **4**, 395 (2006).
6. M. Schmidt, V. de Lorenzo, Synthetic constructs in/for the environment: Managing the interplay between natural and engineered Biology. *FEBS Lett.* **586**, 2199 (2012).
7. C. C. Liu, P. G. Schultz, Adding New Chemistries to the Genetic Code. *An. Rev. Biochem.* **79**, 413 (2010).
8. H. Neumann, K. Wang, L. Davis, M. Garcia-Alai, J. W. Chin, Encoding multiple unnatural amino acids via evolution of a quadruplet-decoding ribosome. *Nature* **464**, 441 (2010).
9. T. S. Young, I. Ahmad, J. A. Yin, P. G. Schultz, An Enhanced System for Unnatural Amino Acid Mutagenesis in E. coli. *Journal of Molecular Biology* **395**, 361 (2009).
10. G. Eggertsson, D. Söll, Transfer ribonucleic acid-mediated suppression of termination codons in Escherichia coli. *Microbiological Reviews* **52**, 354 (September 1, 1988, 1988).
11. F. J. Isaacs, P. A. Carr, H. H. Wang, M. J. Lajoie, B. Sterling, L. Kraal, A. C. Tolonen, T. A. Gianoulis, D. B. Goodman, N. B. Reppas, C. J. Emig, D. Bang, S. J. Hwang, M. C. Jewett, J. M. Jacobson, G. M. Church, Precise Manipulation of Chromosomes in Vivo Enables Genome-Wide Codon Replacement. *Science* **333**, 348 (Jul, 2011).
12. D. G. Gibson, J. I. Glass, C. Lartigue, V. N. Noskov, R. Y. Chuang, M. A. Algire, G. A. Benders, M. G. Montague, L. Ma, M. M. Moodie, C. Merryman, S. Vashee, R. Krishnakumar, N. Assad-Garcia, C. Andrews-Pfannkoch, E. A. Denisova, L. Young, Z. Q. Qi, T. H. Segall-Shapiro, C. H. Calvey, P. P. Parmar, C. A. Hutchison, H. O. Smith, J. C. Venter, Creation of a Bacterial Cell Controlled by a Chemically Synthesized Genome. *Science* **329**, 52 (Jul, 2010).
13. H. H. Wang, F. J. Isaacs, P. A. Carr, Z. Z. Sun, G. Xu, C. R. Forest, G. M. Church, Programming cells by multiplex genome engineering and accelerated evolution. *Nature* **460**, 894 (Aug, 2009).
14. P. A. Carr, H. H. Wang, B. Sterling, F. J. Isaacs, M. J. Lajoie, G. Xu, G. M. Church, J. M. Jacobson, Enhanced multiplex genome engineering through co-operative oligonucleotide co-selection. *Nucleic Acids Res.*, (May 25, 2012, 2012).
15. T. Mukai, A. Hayashi, F. Iraha, A. Sato, K. Ohtake, S. Yokoyama, K. Sakamoto, Codon reassignment in the Escherichia coli genetic code. *Nucleic Acids Res.* **38**, 8188 (2010).
16. D. B. F. Johnson, J. Xu, Z. Shen, J. K. Takimoto, M. D. Schultz, R. J. Schmitz, Z. Xiang, J. R. Ecker, S. P. Briggs, L. Wang, RF1 knockout allows ribosomal incorporation of unnatural amino acids at multiple sites. *Nat Chem Biol* **7**, 779 (2011).

17. K. Ohtake, A. Sato, T. Mukai, N. Hino, S. Yokoyama, K. Sakamoto, Efficient Decoding of the UAG Triplet as a Full-Fledged Sense Codon Enhances the Growth of a prfA-Deficient Strain of Escherichia coli. *J. Bacteriol.* **194**, 2606 (May 15, 2012, 2012).
18. P. O'Donoghue, L. Prat, I. U. Heinemann, J. Ling, K. Odoi, W. R. Liu, D. Söll, Near-cognate suppression of amber, opal and quadruplet codons competes with aminoacyl-tRNAPyl for genetic code expansion. *FEBS Lett.*, (2012).
19. I. U. Heinemann, A. J. Rovner, H. R. Aerni, S. Rogulina, L. Cheng, W. Olds, J. T. Fischer, D. Soll, F. J. Isaacs, J. Rinehart, Enhanced phosphoserine insertion during Escherichia coli protein synthesis via partial UAG codon reassignment and release factor 1 deletion. *FEBS Lett.* **586**, 3716 (2012-Oct-19, 2012).
20. J. T. Ngo, D. A. Tirrell, Noncanonical amino acids in the interrogation of cellular protein synthesis. *Accounts of chemical research* **44**, 677 (2011).
21. H.-S. Park, M. J. Hohn, T. Umehara, L.-T. Guo, E. M. Osborne, J. Benner, C. J. Noren, J. Rinehart, D. Söll, Expanding the Genetic Code of Escherichia coli with Phosphoserine. *Science* **333**, 1151 (August 26, 2011, 2011).
22. R. H. Heineman, I. J. Molineux, J. J. Bull, Evolutionary robustness of an optimal phenotype: Re-evolution of lysis in a bacteriophage deleted for its lysin gene. *J. Mol. Evol.* **61**, 181 (Aug, 2005).
23. J. D. Bain, C. Switzer, R. Chamberlin, S. A. Benner, Ribosome-mediated incorporation of a nonstandard amino acid into a peptide through expansion of the genetic code. *Nature* **356**, 537 (APR 9 1992, 1992).
24. J. C. Anderson, N. Wu, S. W. Santoro, V. Lakshman, D. S. King, P. G. Schultz, An expanded genetic code with a functional quadruplet codon. *Proc. Natl. Acad. Sci. U. S. A.* **101**, 7566 (May 18, 2004, 2004).
25. M. J. Lajoie, S. Kosuri, J. A. Mosberg, C. J. Gregg, D. Zhang, G. M. Church, Probing the limits of genetic recoding in essential genes. *Science* **342**, 361 (2013).
26. S. A. Schwartz, D. R. Helinski, Purification and Characterization of Colicin E1. *J. Biol. Chem.* **246**, 6318 (October 25, 1971, 1971).
27. J. A. Mosberg, M. J. Lajoie, G. M. Church, Lambda Red Recombineering in Escherichia coli Occurs Through a Fully Single-Stranded Intermediate. *Genetics* **186**, 791 (Nov, 2010).
28. D. Figurski, R. Meyer, D. S. Miller, D. R. Helinski, Generation in vitro of deletions in the broad host range plasmid RK2 using phage Mu insertions and a restriction endonuclease. *Gene* **1**, 107 (1976).
29. S. Warming, N. Costantino, D. L. Court, N. A. Jenkins, N. G. Copeland, Simple and highly efficient BAC recombineering using galK selection. *Nucleic Acids Res.* **33**, e36 (2005).
30. N. Rohland, D. Reich, Cost-effective, high-throughput DNA sequencing libraries for multiplexed target capture. *Genome Research* **22**, 939 (May, 2012).
31. W. R. Pearson, T. Wood, Z. Zhang, W. Miller, Comparison of DNA sequences with protein sequences. *Genomics* **46**, 24 (Nov, 1997).
32. B. Langmead, S. L. Salzberg, Fast gapped-read alignment with Bowtie 2. *Nat. Methods* **9**, 357 (Apr, 2012).
33. M. A. DePristo, E. Banks, R. Poplin, K. V. Garimella, J. R. Maguire, C. Hartl, A. A. Philippakis, G. del Angel, M. A. Rivas, M. Hanna, A. McKenna, T. J. Fennell, A. M. Kernytsky, A. Y. Sivachenko, K. Cibulskis, S. B. Gabriel, D. Altshuler, M. J. Daly, A

framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics* **43**, 491 (May, 2011).
34. P. Cingolani, A. Platts, L. L. Wang, M. Coon, N. Tung, L. Wang, S. J. Land, X. Lu, D. M. Ruden, A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w(1118); iso-2; iso-3. *Fly* **6**, 80 (Apr-Jun, 2012).
35. K. Ye, M. H. Schulz, Q. Long, R. Apweiler, Z. Ning, Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics* **25**, 2865 (Nov 1, 2009).
36. K. Chen, J. W. Wallis, M. D. McLellan, D. E. Larson, J. M. Kalicki, C. S. Pohl, S. D. McGrath, M. C. Wendl, Q. Zhang, D. P. Locke, X. Shi, R. S. Fulton, T. J. Ley, R. K. Wilson, L. Ding, E. R. Mardis, BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. *Nat. Methods* **6**, 677 (Sep, 2009).
37. A. R. Quinlan, I. M. Hall, BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26**, 841 (Mar 15, 2010).
38. H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin, P. Genome Project Data, The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**, 2078 (Aug, 2009).
39. D. G. Gibson, H. O. Smith, C. A. Hutchison, J. C. Venter, C. Merryman, Chemical synthesis of the mouse mitochondrial genome. *Nat. Methods* **7**, 901 (Nov, 2010).
40. R. Lutz, H. Bujard, Independent and Tight Regulation of Transcriptional Units in Escherichia Coli Via the LacR/O, the TetR/O and AraC/I1-I2 Regulatory Elements. *Nucleic Acids Res.* **25**, 1203 (March 1, 1997, 1997).
41. D. Wessel, U. I. Flugge, A Method for the Quantitative Recovery of Protein in Dilute-Solution in the Presence of Detergents and Lipids. *Anal. Biochem.* **138**, 141 (1984).
42. A. N. Kettenbach, S. A. Gerber, Rapid and reproducible single-stage phosphopeptide enrichment of complex peptide mixtures: application to general and phosphotyrosine-specific phosphoproteomics experiments. *Anal. Chem.* **83**, 7635 (Oct 15, 2011).
43. A. J. Alpert, Electrostatic repulsion hydrophilic interaction chromatography for isocratic separation of charged solutes and selective isolation of phosphopeptides. *Anal. Chem.* **80**, 62 (Jan 1, 2008).
44. J. V. Olsen, L. M. de Godoy, G. Li, B. Macek, P. Mortensen, R. Pesch, A. Makarov, O. Lange, S. Horning, M. Mann, Parts per Million Mass Accuracy on an Orbitrap Mass Spectrometer via Lock Mass Injection into a C-trap. *Mol Cell Proteomics* **4**, 2010 (Dec, 2005).
45. I. M. Keseler, J. Collado-Vides, A. Santos-Zavaleta, M. Peralta-Gil, S. Gama-Castro, L. Muñiz-Rascado, C. Bonavides-Martinez, S. Paley, M. Krummenacker, T. Altman, P. Kaipa, A. Spaulding, J. Pacheco, M. Latendresse, C. Fulcher, M. Sarker, A. G. Shearer, A. Mackie, I. Paulsen, R. P. Gunsalus, P. D. Karp, EcoCyc: a comprehensive database of Escherichia coli biology. *Nucleic Acids Res.* **39**, D583 (January 1, 2011, 2011).
46. M. A. Shifman, Y. Li, C. M. Colangelo, K. L. Stone, T. L. Wu, K.-H. Cheung, P. L. Miller, K. R. Williams, YPED: A Web-Accessible Database System for Protein Expression Analysis. *Journal of Proteome Research* **6**, 4019 (2007/10/01, 2007).
47. C. A. Schneider, W. S. Rasband, K. W. Eliceiri, NIH Image to ImageJ: 25 years of image analysis. *Nat Meth* **9**, 671 (2012).

48. E. Pennisi, Synthetic Genome Brings New Life to Bacterium. *Science* **328**, 958 (May 21, 2010, 2010).
49. S. Kosuri, N. Eroshenko, E. M. LeProust, M. Super, J. Way, J. B. Li, G. M. Church, Scalable gene synthesis by selective amplification of DNA pools from high-fidelity microchips. *Nat Biotech* **28**, 1295 (2010).
50. M. J. Lajoie, C. J. Gregg, J. A. Mosberg, G. C. Washington, G. M. Church, Manipulating replisome dynamics to enhance lambda Red-mediated multiplex genome engineering. *Nucleic Acids Res.* **40**, e170 (2012-Dec-1, 2012).
51. J. A. Mosberg, C. J. Gregg, M. J. Lajoie, H. H. Wang, G. M. Church, Improving Lambda Red Genome Engineering in *Escherichia coli* via Rational Removal of Endogenous Nucleases. *PLoS One* **7**, e44638 (2012).
52. G. R. Smith, Conjugational Recombination in Escherichia coli: Myths and Mechanisms. *Cell* **64**, 19 (Jan, 1991).
53. R. G. Lloyd, C. Buckman, Conjugational Recombination in Escherichia coli: Genetic Analysis of Recombinant Formation in Hfr X F(-) Crosses. *Genetics* **139**, 1123 (March 1, 1995, 1995).
54. D. B. F. Johnson, C. Wang, J. Xu, M. D. Schultz, R. J. Schmitz, J. R. Ecker, L. Wang, Release Factor One Is Nonessential in Escherichia coli. *ACS Chemical Biology*, (2012).
55. Y. Yamazaki, Niki, H., & Kato, J., Profiling of Escherichia coli Chromosome database. *Methods Mol. Biol.* **416**, 385 (2008).
56. T. Baba, T. Ara, M. Hasegawa, Y. Takai, Y. Okumura, M. Baba, K. A. Datsenko, M. Tomita, B. L. Wanner, H. Mori, Construction of Escherichia coli K-12 in-frame, single-gene knockout mutants: the Keio collection. *Mol. Syst. Biol.* **2**, 11 (2006).
57. P. Funchain, A. Yeung, J. L. Stewart, R. Lin, M. M. Slupska, J. H. Miller, The consequences of growth of a mutator strain of Escherichia coli as measured by loss of function among multiple gene targets and loss of fitness. *Genetics* **154**, 959 (Mar, 2000).
58. R. M. Schaaper, R. L. Dunn, Spectra of spontaneous mutations in Escherichia coli strains defective in mismatch correction: the nature of in vivo DNA replication errors. *Proc. Natl. Acad. Sci. U. S. A.* **84**, 6220 (1987).
59. E. Bi, J. Lutkenhaus, Cell division inhibitors SulA and MinCD prevent formation of the FtsZ ring. *J. Bacteriol.* **175**, 1118 (February 1, 1993, 1993).
60. Y. Ishihama, T. Schmidt, J. Rappsilber, M. Mann, F. U. Hartl, M. J. Kerner, D. Frishman, Protein abundance profiling of the Escherichia coli cytosol. *BMC Genomics* **9**, (Feb 27, 2008).
61. A. Shevchenko, H. Tomas, J. Havlis, J. V. Olsen, M. Mann, In-gel digestion for mass spectrometric characterization of proteins and proteomes. *Nat. Protocols* **1**, 2856 (2007).
62. E. S. Miller, E. Kutter, G. Mosig, F. Arisaka, T. Kunisawa, W. Rüger, Bacteriophage T4 Genome. *Microbiology and Molecular Biology Reviews* **67**, 86 (March 1, 2003, 2003).
63. R. Maldonado, A. J. Herr, Efficiency of T4 Gene 60Translational Bypassing. *J. Bacteriol.* **180**, 1822 (April 1, 1998, 1998).
64. C. E. Jones, T. C. Mueser, N. G. Nossal, Interaction of the Bacteriophage T4 Gene 59 Helicase Loading Protein and Gene 41 Helicase with Each Other and with Fork, Flap, and Cruciform DNA. *J. Biol. Chem.* **275**, 27145 (September 1, 2000, 2000).
65. Y. Chadani, K. Ono, S.-i. Ozawa, Y. Takahashi, K. Takai, H. Nanamiya, Y. Tozawa, K. Kutsukake, T. Abo, Ribosome rescue by Escherichia coli ArfA (YhdL) in the absence of trans-translation system. *Mol. Microbiol.* **78**, 796 (2010).

66. Y. Handa, N. Inaho, N. Nameki, YaeJ is a novel ribosome-associated protein in Escherichia coli that can hydrolyze peptidyl-tRNA on stalled ribosomes. *Nucleic Acids Res.*, (2010).
67. A. Fokine, Z. Zhang, S. Kanamaru, V. D. Bowman, A. A. Aksyuk, F. Arisaka, V. B. Rao, M. G. Rossmann, The Molecular Architecture of the Bacteriophage T4 Neck. *Journal of Molecular Biology* **425**, 1731 (2013).
68. I. J. Molineux, in *The Bacteriophages,* R. Calendar, Ed. (Oxford University Press, New York, 2006), pp. 277-301.
69. P. D. Sadowski, C. Kerr, Degradation of Escherichia coli B Deoxyribonucleic Acid After Infection with Deoxyribonucleic Acid-Defective Amber Mutants of Bacteriophage T7. *J. Virol.* **6**, 149 (August 1, 1970, 1970).
70. L. C. You, P. F. Suthers, J. Yin, Effects of Escherichia coli physiology on growth of phage T7 in vivo and in silico. *J. Bacteriol.* **184**, 1888 (Apr, 2002).

APPENDIX E

Supplemental Material for Probing the Limits of Genetic Recoding in Essential Genes

This supplemental material is reproduced with permission from its initial publication:

Lajoie MJ[*], Kosuri S[*], Mosberg JA, Gregg CJ, Zhang D, Church GM (2013) *Probing the limits of genetic recoding in essential genes*. **Science** 342: 361-3.

Tables and Figures have been renamed to be consistent with CHAPTER 6.

Materials and Methods:

All DNA oligonucleotides (Table S6-8) were purchased with standard purification and desalting from Integrated DNA Technologies. The Oligo Library Synthesis (OLS) array used for synthesizing radically recoded genes was generated on a DNA microchip, processed, and delivered as a ~1-10 pmol lyophilized pool of oligos by Agilent Technologies (Carlsbad, CA).

Cultures were grown at 34 °C with aeration in LB-Lennox (LB^L; 10 g/L Bacto tryptone, 5 g/L sodium chloride, 5 g/L yeast extract) and colonies were grown on LB^L-agar plates (LB^L with 15 g/L Bacto agar). LB^L media was supplemented with one or more of the following selective agents: carbenicillin (50 μg/mL), sodium dodecyl sulfate (SDS; 0.005% w/v), kanamycin (30 μg/mL). Colicin E1 was obtained via expression in strain JC411 (*31*), and purified as previously described.

NAT_*kan*^{R} cassette preparation

Kanamycin resistance (*kan*^{R}) cassettes were inserted *via* λ Red recombination (*32*, *33*) downstream of essential ribosomal genes, in order to test whether polar effects from inserting *kan*^{R} impair fitness. These "NAT_*kan*^{R}" cassettes were PCR amplified using primers that introduced 50 bp of genomic homology on either side of the intended *kan*^{R} insertion site (Kapa HiFi Ready Mix; manufacturer's protocols). PCR products were SPRI purified as previously described (*34*), eluted in deionized water (dH_2O), and checked on a 1% agarose gel for correct size and purity before being recombined as described below.

Recoded gene cassette preparation

Recoded essential genes (Table S6-9) were synthesized from an Agilent OLS array as previously described (*35*). Due to their size, the *prfB* and *rpsA* genes were difficult to synthesize in one piece, so they were each synthesized in two pieces, which were then assembled *via* isothermal assembly (*36*). All synthesized recoded cassettes were fused to a downstream kanamycin resistance gene (*kan*^{R}) *via* isothermal assembly (*36*). The crude isothermal assemblies were PCR amplified using primers (Table S6-8) that introduced 50 bp of genomic homology on either side of the recoded gene and *kan*^{R} (Kapa HiFi Ready Mix; manufacturer's protocols). Full-length cassettes were SPRI purified as previously described (*34*), eluted in dH_2O, and checked on a 1% agarose gel for correct size and purity before being recombined as described below.

Partially recoded cassette preparation

Partially recoded gene cassettes were prepared using the full-length recoded gene cassettes (described above) as template (Kapa HiFi Ready Mix; manufacturer's protocols). While the same reverse primers were used, new forward primers were designed to hybridize inside the recoded cassette and to introduce 50 bp homology regions matching the natural sequence, so that only the C-terminal portion of the gene would be recoded (Figure 6-1B).

We prepared two types of partially recoded cassettes. The less stringent version recoded exactly half of the gene. The more stringent version recoded all except for the first 30 codons of the gene. Partially recoded cassettes were SPRI purified as previously described (*34*), eluted in

dH_2O, and checked on a 1% agarose gel for correct size and purity before being recombined as described below.

CoS-MAGE selectable marker preparation

To maximize the number of alleles that could simultaneously be replaced per recombinant, we used Co-Selection Multiplex Automated Genome Engineering (CoS-MAGE) with *tolC* or *bla* as co-selectable markers (*37*, *38*). In most cases, 90 nt MAGE oligos were designed to replace several forbidden codons. We performed CoS-MAGE in an EcNR2.*xseA*$^-$ background, which has ExoVII inactivated in order to minimize allele loss near the 3' end of the MAGE oligos (*39*). Since the ribosomal genes are clustered in different regions of the genome, selectable markers needed to be placed in multiple different genomic locations in order to provide co-selection in adequate proximity (~500 kb) to the target ribosomal genes. Therefore, we prepared two *tolC* cassettes (*tolC*.3502900 for *rpsL*, *rplQ*, *rplO*, *rpsG*, *rplD*, *rpsD*, *rpsC*, and *rplB*; *tolC*.4427600 for *rpsR*, *rplL*, and *rplJ*) using Kapa HiFi Ready Mix (manufacturer's protocols) and PCR primers that introduced 50 bp of flanking genomic homology (Table S6-8). The *tolC* cassettes were purified using Qiagen's PCR purification kit (manufacturer's protocols, eluted in dH_2O) before being recombined as described in the "gene and allele replacement" methods section. For *rpsA* co-selection, *bla* was already present in the λ prophage of EcNR2.

Gene and allele replacement

All CoS-MAGE oligonucleotides and Nat_*Kan*^{R}, fully recoded, and partially recoded cassettes (described above) were recombined into EcNR2 (*E. coli* MG1655 Δ*mutS*::*cat* Δ(*ybhB-bioAB*)::[λcI857 N(*cro-ea59*)::*tetR-bla*]) as previously described (*38*). Briefly, EcNR2 was grown to mid-log phase (OD_{600} between 0.4 and 0.6), induced to express λ Red for 15 minutes in a 42 °C shaking water bath, and chilled on ice. For each recombination, 1 mL of induced culture was washed twice in 1 mL cold dH_2O, and then the cell pellet was resuspended in 50 µL of dH_2O containing the DNA to be recombined. For PCR products, 1-2 ng/µL was used; to inactivate selectable markers for CoS-MAGE, a 90mer oligonucleotide was used at a final concentration of 1 µM; for CoS-MAGE, 90mer oligonucleotides were pooled at a final concentration of ≤ 5 µM. A BioRad GenePulser™ was used for electroporation (0.1 cm cuvette, 1.78 kV, 200 Ω, 25 µF), and electroporated cells were allowed to recover in 3 ml LB^L in a rotator drum at 34°C for at least 3 hours before plating on appropriate selective media.

Recombinant clones were selected on LB^L-agar supplemented with kanamycin, and then re-streaked on fresh LB^L-agar supplemented with kanamycin to ensure monoclonality. Monoclonal colonies were then grown in a 96-well format (150 µL LB^L supplemented with kanamycin) in preparation for genetic analysis.

To prepare the EcNR2.*xseA*$^-$ strains for CoS-MAGE, we deleted the endogenous *tolC* from the genome using the tolC.90.del oligo and selected for recombinants *via* Colicin E1 selection (*38*). We then separately introduced the *tolC* co-selection cassettes (one per CoS-MAGE strain) and selected on LB^L supplemented with SDS. Finally, we inactivated *tolC* by introducing a nonsense mutation and a frameshift using the tolC-r_null_mut* oligo. For *bla* co-selection, we used the

bla_mut* oligo to inactivate *bla* (present in the λ prophage) and screened for carbenicillin-sensitive recombinants by replica plating on LB^L supplemented with carbenicillin.

CoS-MAGE: CoS-MAGE was performed as previously described (*37*), using 0.5 μM of each MAGE oligo and 0.5 μM of the appropriate co-selection oligo to revert *tolC*.3502900 (*rpsL*, *rplQ*, *rplO*, *rpsG*, *rplD*, *rpsD*, *rpsC*, *rplB*), *tolC*.4427600 (*rpsR*, *rplL*, *rplJ*), or *bla* (*rpsA*). MAGE (without co-selection) (*40*) was performed on *rpsP* and *rpsB* because they were distant from the available co-selectable markers and only had 4 codons to be removed. CoS-MAGE recombinants were selected on LB^L-agar supplemented with SDS (for *tolC*) or LB^L-agar supplemented with carbenicillin (for *bla*), and MAGE recombinants were grown on LB^L-agar without selection. Monoclonal colonies were picked into a 96-well plate and grown under the appropriate selection at 34 °C with shaking.

Recombinant clone genotyping

Recombinant clones were first screened by PCR, then validated by Sanger sequencing.

PCR screens: For the fully recoded genes, we performed 3 PCR reactions for each clone. As diagramed in Figure 6-1B, the three sets of primers hybridized to the natural gene sequence (NAT), the recoded gene sequence (SYN), and the flanking genomic region (BND). PCR reactions (10 μL each) were performed with Kapa 2G Fast Ready Mix according to the manufacturer's protocols. Adequate primer specificity was observed with a 58 °C annealing temperature. Desired recombinants had no NAT amplicon, a gene-sized SYN amplicon, and a BND amplicon 847 bp larger than that of the wild type negative control. Partially (C-terminally) recoded recombinants were screened using the NAT forward and SYN reverse primers (desired recombinants had a gene-sized amplicon) and BND primers (desired recombinants showed an 847 bp increase in amplicon size). All putative recombinants that passed the PCR assay were Sanger sequenced (Genewiz or Eton Bioscience Inc.) using the forward BND primers and/or kanR.seqOUT-Nr2.

CoS-MAGE recombinants were typically sequenced without initial Multiplex Allele Specific Colony PCR (MASC-PCR (*38*)) screening because the targeted alleles were too close together to allow for the amplification of discrete bands. However, well-separated alleles were screened via MASC-PCR with standard protocols (*38*) prior to Sanger sequencing validation.

Doubling time analysis

Doubling times (Figure 6-2, Tables S6-4 to S6-5) were determined for all recoded clones using LB^L and Teknova HiDef Azure media. Kinetic growth curves were performed in triplicate on a Biotek H4 plate reader with OD_{600} measurements at 5 minute intervals. Cultures were grown in a flat-bottom 96-well plate (in 150 μL of LB^L supplemented with carbenicillin) with shaking at 34 °C. Doubling times were determined by $t_{double} = c*\ln(2)/m$, where c = 5 minutes per time point and m is the maximum slope of $\ln(OD_{600})$ smoothed across 5 contiguous time points (20 minutes). We typically calculate doubling time in this manner so as to accommodate strains that achieve lower maximum optical densities. Each data point in Figure 6-2 represents the average doubling time of an individual strain with one ribosomal gene partially or fully recoded (n = 3).

Each replicate was prepared by passaging from the previous one. All strains are based on EcNR2 or EcNR2.xseA$^-$ (doubling times under assay conditions for these strains are 49 +/- 4 minutes in LB^L and 84 +/- 5 minutes in Teknova HiDef Azure Media (12 replicates per condition)).

Supplemental information:

Design parameters for radically recoded genes

- We removed all instances of 13 rare codons (UAG, AGA, AGG, CUU, CUC, CCC, ACC, AUA, GUC, GCC, UCC, CGG, UGA) by replacing them with synonymous codons. Since our goal is to radically change the genetic code, codon removal is merely the first step toward removing and/or reassigning anticodon function, and all codons uniquely recognized by a tRNA or release factor must be changed prior to deletion. Therefore, rather than choosing the 13 rarest codons, we instead targeted codons that are recognized by the least frequently used anticodons. Removing all instances of these codons from the genome would permit the deletion of 10 anticodons (three less than the 13 codons removed, as both CUC and CUU correspond to the same Leu anticodon, both AGG and AGA correspond to the same Arg anticodon, and RF2 is still necessary to terminate UAA in the absence of RF1) and the introduction of 4 nonstandard amino acids into the genetic code (AUA, UAG, CGG, and AGA/AGG codons can be unambiguously reassigned to encode a new amino acid; the introduction of tRNAs for the other codons would cause ambiguous amino acid incorporation due redundant anticodon specificities – see Figure 6-1A).
- All start codons were changed to AUG (*rpsM* GUG→AUG).
- All non-forbidden codons in radically recoded genes (blue segments in Figure 6-3) were swapped with a synonymous codon to reduce nucleotide identity to the natural sequence (see *rpmH* example below, page 4 of SOM). We randomly chose the synonymous replacement codon from a weighted distribution of the remaining possible codons based on their frequency in the *E. coli* MG1655 genome.
- Genes that encoded overlapping coding DNA sequences were modified to remove these overlaps. If another gene overlapped at the start of the coding sequence, the end of this gene was duplicated, and the start codon was removed. If the start codon could not be removed, an in-frame stop-codon was added to prevent translation initiation from the undesired start codon. If a gene overlapped at the end of the coding sequence, we removed the start codon from the recoded sequence. We ensured that the subsequent gene was still translated by duplicating the natural sequence downstream of the recoded sequence. Table S6-10 provides a list of these refactored overlaps.
- Genetically encoded frameshifts were removed (*prfB* CUUU73CUU).
- The following restriction sites were removed: BtsI, BsaI, BsmBI, BspQI, XbaI, and AatII.
- The mRNA secondary structure near the ribosomal binding site was minimized. To accomplish this, we used UNAFold (*41*) to calculate the ΔG for the secondary structure of a 42 bp window centered at the translation start site. The initial design was optimized in order to reduce secondary structure if one of the following two conditions were met: (1) the recoded secondary structure was stronger than the original secondary structure and less than ΔG -7.0 kcal/mol, or (2) the recoded secondary structure was less than ΔG = -10 kcal/mol. To optimize the recoded sequence, all available synonymous codons were varied individually and a new sequence with reduced secondary structure was selected.

For example, below is the comparison between the natural and recoded sequences for *rpmH*. Nucleotide abbreviations are according to IUPAC notation. Yellow highlighting indicates nucleotides that differ in the recoded gene.

>radical recoding *rpmH*
ATG AAR CGY ACK TTY CAR CCK WST GTW YTG AAR CGY AAY CGY TCW CAY
GGY TTY CGY GCK CGY ATG GCW ACK AAR AAY GGY CGY CAR GTK YTG GCR
CGY CGY CGY GCW AAR GGY CGY GCK CGY YTR ACS GTK TCW AAR TAA

Partial replacement with full-length recoded cassettes
Fully recoded *rpmH* and *rplT* cassettes repeatedly produced partially recoded recombinants (wild type until C81 and T147, respectively). In both cases, the position of the crossover was shifted upstream by using partially recoded constructs that preserved the wild type N-termini (Figure S6-2). This indicates that the full-length cassette 1) had a lethal design element in its N-terminus and/or 2) had poorly recominogenic homology sequence.

Double mutants with full-length recoded cassettes
To understand the effect of combining multiple recoded genes in a single strain, we transcriptionally fused the recoded rplM_syn1 variant (third slowest doubling time) or rpsI_syn1 variant (fourth slowest doubling time) to a spectinomycin resistance gene, generated double mutants in rplP_syn1 (slowest doubling time and contains ATA forbidden codon), rpmC_syn1 (second slowest doubling time), and rplE_syn1 (normal doubling time), and selected the highest fitness recombinant exhibiting the desired genotype. When grown in LB^L without antibiotic supplementation, all double mutants grew faster than expected assuming additive fitness defects for independent mutations (Figure S6-1, Table S6-11). It is possible that compensatory off-target mutations facilitated by inactive mismatch repair may alleviate growth impairment, and double mutants may exhibit varying fitness effects due to ribosomal protein autoregulation (*42*).

Remaking rplP_syn2, rpsS_syn2, and rpmD_syn2
We re-sequenced all gene replacement strains (Table S6-4) to confirm that no mutations had occurred during extended growth. We observed a G36A mutation in rplP_syn1, which introduced a forbidden AUA codon; a C5T mutation in rpsS_syn1, which introduced a forbidden CUU codon; and a putative duplication in rpmD_syn1, resulting in the presence of both a natural and recoded copy of *rpmD* in the same genome.

For *rplP*, the wild type AUG codon resulted in an extreme fitness disadvantage, which provided a strong selection for the spontaneous G36A mutation (AUG→AUA change). Therefore, we attempted to change the forbidden AUA codon to all other Ile (AUC and AUU) and Met (AUG) codons using MAGE. While AUG was not observed, presumably due to an extreme fitness disadvantage, AUU and AUC were well-tolerated, leading to rplP_syn2. Since this mutation was intended, it was not counted as a synthesis error or represented by a yellow line in Figure 6-3.

For *rpsS*, resequencing revealed a forbidden CUU codon. We re-amplified the *rpsS* gene with primers that changed this forbidden CUU to all permitted Pro (CCA, CCG, and CCU) and Leu (UUA, UUG, CUA, and CUG) codons. Codons CCA and CCG (but not CCU) were observed for Pro. Codons UUA, UUG, CUA, and CUG were observed for Leu. Additionally, we allowed the

subsequent CGC Codon be shuffled to CGA, CGU, and CGC. We selected a clone, rpsS_syn2, with intended mutations T6A and C9A. Since these mutations were intended, they were not counted as synthesis errors or represented by yellow lines in Figure 6-3.

For *rpmD*, we repeated the insertion of the original synthetic *rpmD* gene, yielding a clone, rpmD_syn2, with the correct genotype.

Supplemental Figures:

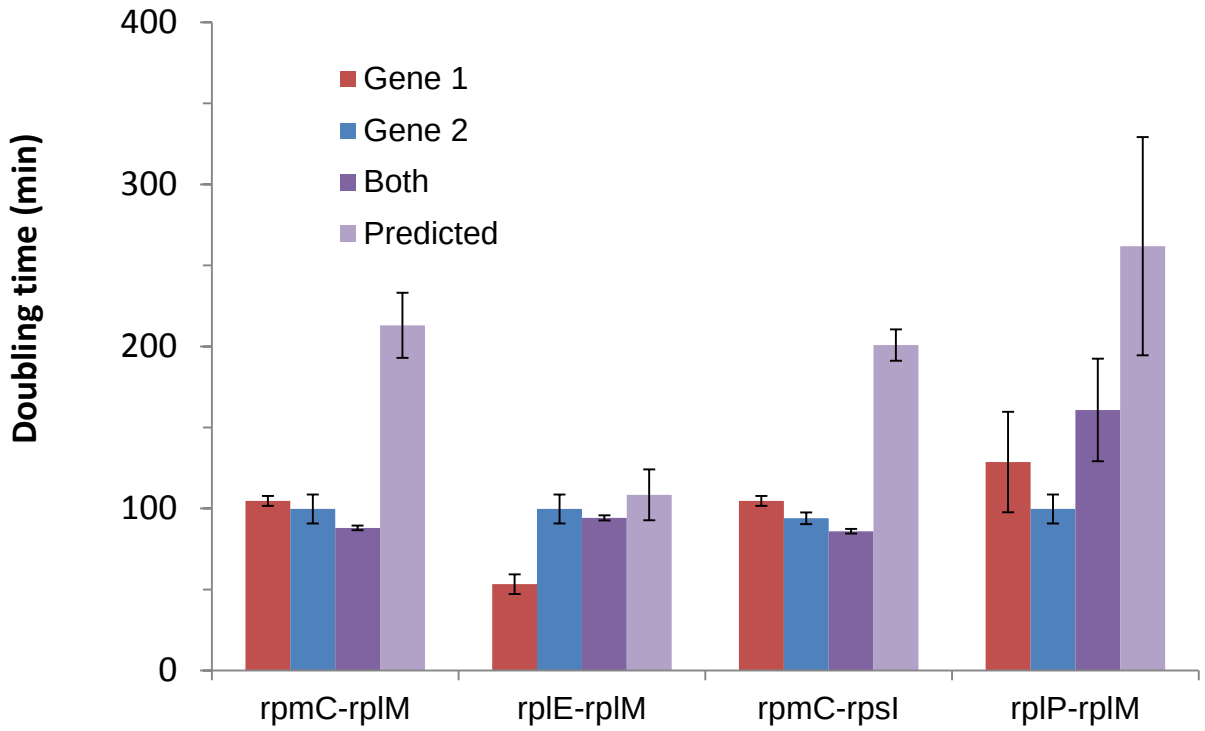


Figure S6-1. Doubling times of double mutants compared to single mutants. Synthetic gene 2 (*rplM* or *rpsI*) was transcriptionally fused to a spectinomycin resistance gene and recombined into strains rpmC_syn1, rplE_syn1, and rplP_syn1. Double mutants that were resistant to both kanamycin and spectinomycin were isolated, Sanger sequence verified, and assayed for doubling time in LB^{L} without antibiotic supplementation. The double mutant exhibiting the fastest doubling time for each gene pair was chosen. Data bars represent the doubling times of the gene 1 syn strain (red), gene 2 syn strain (blue), double mutant (dark purple), and predicted doubling time of the double mutant assuming that fitness defects are independent (light purple). Error bars are the standard deviation of 3 technical replicates; variances for the predictions are the square root of the summed squared variances of the two measured strains.

Supplemental Tables:

Table S6-1. Recoded essential gene design attributes
Table S6-2. Genes with forbidden codons successfully removed after each phase of recoding
Table S6-3. Forbidden codons remaining after each phase of recoding
Table S6-4. Gene replacement strain summary and doubling times
Table S6-5. CoS-MAGE strain summary and doubling times
Table S6-6. Oligos used to replace rplQ CUU 160-162
Table S6-7. Successful rplQ CUU 160-162 replacements
Table S6-8. Primers and oligonucleotides used in this study
Table S6-9. Recoded gene designs
Table S6-10. Refactored overlapping genes
Table S6-11. Doubling times of double mutants compared to single mutants

Table S6-1. Recoded essential gene design attributes

Gene	Length	Forbidden codons	Number of changes	% nt identity	Total codons	Identical codons	Changed codons	% codon identity
rpmH	141	1	46	67.4%	47	3	44	6.38%
rpmD	180	4	59	67.2%	60	5	55	8.33%
rpmC	192	2	71	63.0%	64	3	61	4.69%
rpsR	228	6	84	63.2%	76	2	74	2.63%
rpmB	237	7	84	64.6%	79	4	75	5.06%
rpsP	249	1	86	65.5%	83	3	80	3.61%
rpsQ	255	5	88	65.5%	85	4	81	4.71%
rpmA	258	3	92	64.3%	86	2	84	2.33%
rpsS	279	6	96	65.6%	93	5	88	5.38%
rplW	303	10	101	66.7%	101	6	95	5.94%
rpsN	306	6	106	65.4%	102	7	95	6.86%
rplU	312	3	105	66.3%	104	5	99	4.81%
rpsJ	312	10	104	66.7%	104	3	101	2.88%
rplX	315	10	111	64.8%	105	4	101	3.81%
rplV	333	8	119	64.3%	111	3	108	2.70%
rplS	348	8	127	63.5%	116	4	112	3.45%
rplR	354	10	124	65.0%	118	2	116	1.69%
rplT	357	3	126	64.7%	119	4	115	3.36%
rpsM	357	13	119	66.7%	119	4	115	3.36%
rplL	366	1	122	66.7%	122	7	115	5.74%
rplN	372	10	132	64.5%	124	6	118	4.84%
rpsL	375	9	131	65.1%	125	3	122	2.40%
rplQ	384	4	138	64.1%	128	5	123	3.91%
rpsK	390	8	130	66.7%	130	5	125	3.85%
rpsH	393	13	135	65.6%	131	8	123	6.11%
rpsI	393	8	140	64.4%	131	6	125	4.58%
rplP	411	7	137	66.7%	137	9	128	6.57%
rplM	429	7	134	68.8%	143	10	133	6.99%
rplO	435	5	152	65.1%	145	7	138	4.83%
rplJ	498	11	176	64.7%	166	7	159	4.22%
rpsE	504	15	169	66.5%	168	9	159	5.36%
rplF	534	12	182	65.9%	178	5	173	2.81%
rplE	540	14	186	65.6%	180	8	172	4.44%
rpsG	540	14	192	64.4%	180	8	172	4.44%
rplD	606	11	210	65.3%	202	10	192	4.95%
rpsD	621	12	224	63.9%	207	6	201	2.90%
rplC	630	14	216	65.7%	210	8	202	3.81%
rpsC	702	9	237	66.2%	234	11	223	4.70%
rpsB	726	12	250	65.6%	242	14	228	5.79%
rplB	822	13	284	65.5%	274	10	264	3.65%
prfB	1098	46	388	64.7%	366	17	349	4.64%
rpsA	1674	34	583	65.2%	558	17	541	3.05%
Total	**18759**	**405**	**6496**	**65.4%**	**6253**	**269**	**5984**	**4.44%**

Table S6-2. Genes with forbidden codons successfully removed after each phase of recoding

Gene	Operon location[a]	*Kan*^{R} only[b]	Full cassettes[c]	Partial cassettes[d]	CoS-MAGE
rpmH	Start				
rpmD	Middle				
rpmC	Middle				
rpsR	Middle			5/6	
rpmB	Complex				
rpsP	Start			0/1	
rpsQ	End				
rpmA	End				
rpsS	Middle				
rpsN	Middle				
rplU	Start				
rpsJ	Start				
rplX	Middle				
rplW	Middle				
rplV	Middle				
rplS	End				
rplR	Middle				
rplT	Complex				
rpsM	Start				
rplL	Complex			0/1	
rplN	Start				
rpsL	Start			6/9	
rplQ	End				
rpsK	Middle				
rpsI	End				
rpsH	Middle				
rplM	Start				
rplP	Middle				
rplO	Middle	Not observed			
rplJ	Complex				
rpsE	Middle				
rplF	Middle				
rplE	Middle				
rpsG	Middle				
rplD	Middle				
rpsD	Middle			7/12	
rplC	Middle				
rpsC	Middle			3/9	
rpsB	Middle			9/12	
rplB	Middle				
prfB	Complex				
rpsA	Complex				

[a] Start = first ORF in operon, Middle = flanked on both sides by other ORFs in same operon, End = last ORF in operon, Complex = multiple overlapping transcriptional units

[b] Purple indicates successful insertion of *kan*^{R} into the operon without recoding

[c] Dark green indicates genes with all forbidden codons removed during that phase

[d] Lime green indicates genes that had all forbidden codons removed in a previous phase; light green indicates genes with a subset of their forbidden codons removed (instances removed/total instances) by partially recoded cassettes

Table S6-3. Forbidden codons remaining after each phase of recoding

Codon removed	Natural assignment	Instances in genome	Instances in targeted essential genes	Fully recoded cassettes[a]	Partially recoded cassettes[b]	CoS-MAGE[c]
UAG	STOP (RF1)	321	0	0	0	0
AGA/AGG	Arg	4,228	5	1	1	0
CUU/CUC	Leu	30,030	50	19	14	0*
CCC	Pro	7,401	3	1	1	0
ACC	Thr	31,766	133	53	37	0
AUA	Ile	5,797	1	0	0	0
GUC	Val	20,757	59	12	10	0
GCC	Ala	34,747	65	22	18	0
UCC	Ser	11,672	82	35	27	0
CGG	Arg	7,273	3	1	1	0
UGA	STOP (RF2)	1,232	4	2	2	0
	Total remaining	**155,224**	**405**	**146**	**111**	**0**

[a] Instances of each forbidden codon remaining after recombination with fully recoded cassettes
[b] Instances of each forbidden codon remaining after recombination with partially recoded cassettes
[c] Instances of each forbidden codon remaining after CoS-MAGE
*Original desired *rplQ* U162G (CUU→CUG) change was not observed, but this was overcome using diversity (see Table S6-7)

Table S6-4. Gene replacement strain summary and doubling times

							NAT_*Kan*^{R} LB^L doubling time (min)[c]			NAT_*Kan*^{R} Azure doubling time (min)[c]			SYN_*Kan*^{R} LB^L doubling time (min)[d,e]			SYN_*Kan*^{R} Azure doubling time (min)[d,e]		
Strain name[a]	**Gene**	**Switch-over index**[b]	**Forbidden codons removed**	**Unintended mismatches**	**Unintended deletions**	**Total mutations**	**Rep1**	**Rep2**	**Rep3**	**Rep1**	**Rep2**	**Rep3**	**Rep1**	**Rep2**	**Rep3**	**Rep1**	**Rep2**	**Rep3**
rpmH_chim1	rpmH	69	1	0	0	17	56	60	59	93	102	104	52	52	48	80	77	77
rpmD_syn1	rpmD	0	4	0	0	59	75	77	76	111	122	130	55	57	46	113	117	116
rpmD_syn2	rpmD	0	4	0	0	59	75	77	76	111	122	130	89	94	91	169	175	171
rpmC_syn1	rpmC	0	2	0	0	71	55	56	55	105	102	83	108	104	102	128	110	117
rpsR_chim1	rpsR	90	5	0	0	54	49	50	52	91	90	79	78	61	61	95	78	92
rpmB_syn1	rpmB	0	7	2	0	84	51	49	51	106	87	85	58	59	63	137	131	137
rpsP_chim1	rpsP	92	0	0	0	54	49	49	50	105	79	83	54	53	61	104	115	104
rpsQ_syn1	rpsQ	0	5	0	0	88	61	60	55	103	98	69	70	68	65	102	97	93
rpmA_syn1	rpmA	0	3	1	0	91	47	48	47	80	78	79	66	66	74	85	93	90
rpsS_syn1	rpsS	0	6	1	0	96	51	51	53	100	105	97	64	57	63	123	97	99
rpsS_syn2	rpsS	0	6	0	0	96	51	51	53	100	105	97	49	50	50	80	79	76
rplW_syn1	rplW	0	10	2	0	102	42	54	52	114	110	109	67	69	61	113	111	124
rpsN_syn1	rpsN	0	6	3	0	107	52	52	53	86	92	84	68	69	69	69	91	89
rplU_syn1	rplU	0	3	0	0	105	49	49	49	85	89	65	74	74	74	140	129	138
rpsJ_syn1	rpsJ	0	10	1	0	105	54	55	55	77	81	73	59	61	63	82	113	107
rplX_syn1	rplX	0	10	0	0	111	91	93	88	121	117	113	57	55	53	105	87	83
rplV_syn1	rplV	0	8	0	0	119	52	53	52	97	101	90	90	69	61	119	107	119
rplS_syn1	rplS	0	8	2	0	127	48	50	51	97	87	84	57	58	56	105	120	90
rplR_syn1	rplR	0	10	3	0	125	51	50	50	57	80	75	54	53	47	87	96	79
rplT_chim1	rplT	92	3	1	0	96	61	56	56	104	96	96	69	67	66	98	94	88
rpsM_syn1	rpsM	0	13	1	1	119	56	57	59	92	88	81	69	62	62	95	85	104
rplL_chim1	rplL	90	1	0	0	94	63	63	61	92	110	106	54	56	51	90	88	68

Table S6-4 (Continued).																		
rplN_syn1	rplN	0	10	2	0	132	61	63	53	60	77	61	49	51	48	92	70	90
rpsL_chim1	rpsL	186	6	1	1	65	56	54	57	71	83	80	75	78	77	103	101	95
rpsK_syn1	rpsK	0	8	0	0	130	54	52	54	79	87	93	95	79	83	107	111	95
rpsI_syn1	rpsI	0	8	1	0	141	51	51	42	92	90	85	95	97	90	104	107	124
rpsH_syn1	rpsH	0	13	0	0	135	53	55	50	101	77	78	64	63	58	97	86	99
rplP_syn1	rplP	0	7	1	0	137	58	59	54	93	92	110	164	116	106	158	112	158
rplP_syn2	rplP	0	7	0	0	137	58	59	54	93	92	110	67	65	61	134	136	138
rplM_syn1	rplM	0	7	1	0	135	58	59	52	95	98	83	110	94	95	141	149	134
rpsE_syn1	rpsE	0	15	0	0	169	58	62	50	111	134	95	51	54	51	108	105	102
rplF_syn1	rplF	0	12	1	0	183	71	75	57	139	134	153	60	84	72	105	110	89
rplE_syn1	rplE	0	14	0	0	186	60	65	46	123	142	125	47	59	54	101	81	89
rpsD_chim1	rpsD	90	7	0	0	190	70	72	67	98	77	93	74	72	66	119	120	130
rplC_syn1	rplC	0	14	1	0	216	58	61	58	104	88	83	73	77	97	151	127	127
rpsC_chim1	rpsC	351	3	0	1	118	57	58	54	93	82	105	60	58	57	129	109	164
rpsB_chim1	rpsB	90	9	2	0	225	50	45	45	93	87	92	46	53	48	90	79	69
prfB_syn1	prfB	0	46	1	0	389	55	57	55	86	91	75	56	60	49	110	97	82
	Total:	**294**	**26**	**3**	**4375**													

[a] Strains are named for their recoded genes; "syn" indicates fully recoded; "chim" indicates partially recoded; the original "syn" strains rpmD_syn1, rpsS_syn1, and rplP_syn1 gained forbidden codons, so an additional clone was generated and characterized for each gene (rpmD_syn2, rpsS_syn2, and rplP_syn2). Although the original rpmD_syn1, rpsS_syn1, and rplP_syn1 strains are still reported in gray letters, their forbidden codons removed, unintended mismatches, and unintended deletions were not included in the totals at the bottom of the table.

[b] Beginning of radically recoded portion in partially recoded genes

[c] NAT_*Kan*^{R} indicates that a kanR gene was inserted without recoding the target gene

[d] SYN_*Kan*^{R} indicates that the target gene was radically recoded

[e] Some doubling times appear to decrease across subsequent replicates, possibly indicating that spontaneous mutagenesis improves fitness. We note that each strain was passaged at least twice prior to sequence verification.

Table S6-5. CoS-MAGE strain summary and doubling times

				NAT_*Kan*^{R} LB^L doubling time (min)[c]			NAT_*Kan*^{R} Azure doubling time (min)[c]			CoS-MAGE LB^L doubling time (min)			CoS-MAGE Azure doubling time (min)		
Strain name[a]	**Gene**	**Alleles targeted**	**Alleles converted**[b]	**Rep1**	**Rep2**	**Rep3**	**Rep1**	**Rep2**	**Rep3**	**Rep1**	**Rep2**	**Rep3**	**Rep 1**	**Rep 2**	**Rep 3**
rpsR_CoS1	rpsR	1	1	49	50	52	91	90	79	50	50	51	58	72	56
	rpsP	1	1	49	49	50	105	79	83						
	rpsB	3	3	50	45	45	93	87	92						
rpsR_CoS2	rpsR	1	1	49	50	52	91	90	79	50	50	53	60	72	61
	rplL	1	1	63	63	61	92	110	106						
	rpsB	3	3	50	45	45	93	87	92						
rpsL_CoS1	rpsL	3	3	56	54	57	71	83	80	49	49	49	44	64	62
rplQ_CoS1	rplQ	4	1	64	66	66	113	109	100	54	50	50	110	90	79
rplQ_CoS2	rplQ	4	1	64	66	66	113	109	100	53	49	50	62	92	74
rplQ_CoS3	rplQ	4	1	64	66	66	113	109	100	49	49	48	80	72	78
rplQ_CoS4	rplQ	4	1	64	66	66	113	109	100	50	50	47	65	80	68
rplQ_CoS5	rplQ	4	1	64	66	66	113	109	100	48	48	48	85	85	71
rplQ_CoS6	rplQ	4	1	64	66	66	113	109	100	49	48	48	75	72	86
rplQ_CoS7	rplQ	4	1	64	66	66	113	109	100	51	50	48	87	74	65
rplQ_CoS8	rplQ	4	1	64	66	66	113	109	100	48	49	48	76	92	92
rplQ_CoS9	rplQ	4	1	64	66	66	113	109	100	50	49	48	69	76	92
rplQ_CoS10	rplQ	4	1	64	66	66	113	109	100	53	52	52	106	95	96
rplQ_CoS11	rplQ	4	1	64	66	66	113	109	100	53	49	50	78	74	40
rplQ_CoS12	rplQ	4	1	64	66	66	113	109	100	49	54	53	80	116	114
	rpsC	6	2	57	58	54	93	82	105						
rplO_CoS1	rplO	5	2	nr	nr	nr	nr	nr	nr	46	51	49	101	91	100
rplO_CoS2	rplO	5	2	nr	nr	nr	nr	nr	nr	48	51	50	72	66	72
rplO_CoS3	rplO	5	1	nr	nr	nr	nr	nr	nr	48	54	51	70	65	72
rplJ_CoS1	rplJ	11	3	51	51	49	114	87	90	48	49	49	67	58	49
rplJ_CoS2	rplJ	11	3	51	51	49	114	87	90	60	57	56	84	59	56
rplJ_CoS3	rplJ	11	3	51	51	49	114	87	90	53	51	52	89	102	87

Table S6-5 (Continued).															
rplJ_CoS4	rplJ	11	3	51	51	49	114	87	90	52	51	52	83	68	62
rplJ_CoS5	rplJ	11	2	51	51	49	114	87	90	55	51	53	88	78	68
rplJ_CoS6	rplJ	11	2	51	51	49	114	87	90	51	48	49	70	78	68
rplJ_CoS7	rplJ	11	1	51	51	49	114	87	90	49	51	54	93	88	99
	rpsA	34	1	48	50	44	93	95	100						
rpsG_CoS1	rpsG	14	10	49	50	44	84	83	84	51	53	52	61	79	97
rpsG_CoS2	rpsG	14	7	49	50	44	84	83	84	51	53	53	76	82	63
rpsG_CoS3	rpsG	14	7	49	50	44	84	83	84	48	53	52	86	106	72
rpsD_CoS1	rpsD	5	3	70	72	67	98	77	93	52	55	56	98	100	99
rpsD_CoS2	rpsD	5	3	70	72	67	98	77	93	56	51	51	84	94	85
rplD_CoS1	rplD	11	2	52	53	54	104	108	105	51	50	52	97	77	60
rplD_CoS2	rplD	11	4	52	53	54	104	108	105	49	53	49	86	92	90
rplD_CoS3	rplD	11	7	52	53	54	104	108	105	48	52	51	51	78	73
rplD_CoS4	rplD	11	1	52	53	54	104	108	105	49	48	49	55	83	86
	rpsA	34	1	48	50	44	93	95	100						
rpsC_CoS1	rpsC	6	4	57	58	54	93	82	105	52	52	49	102	85	70
rpsC_CoS2	rpsC	6	4	57	58	54	93	82	105	50	54	54	62	56	57
rplB_CoS1	rplB	13	3	50	53	45	88	75	87	50	54	54	73	58	59
rplB_CoS2	rplB	13	1	50	53	45	88	75	87	61	63	63	84	95	93
rplB_CoS3	rplB	13	4	50	53	45	88	75	87	50	56	50	71	77	65
rplB_CoS4	rplB	13	7	50	53	45	88	75	87	50	54	51	76	50	68
rpsA_CoS1	rpsA	34	19	48	50	44	93	95	100	45	50	43	76	53	61
rpsA_CoS2	rpsA	34	7	48	50	44	93	95	100	47	50	44	90	79	91
rpsA_CoS3	rpsA	34	12	48	50	44	93	95	100	47	51	44	103	84	77
rpsA_CoS4	rpsA	34	16	48	50	44	93	95	100	46	51	43	89	85	89
rpsA_CoS5	rpsA	34	15	48	50	44	93	95	100	45	49	45	88	90	89

[a] 45 total CoS-MAGE strains

[b] 187 total forbidden codons (111 unique positions) removed across 45 CoS-MAGE strains

[c] nr indicates that no recombinants were observed

Table S6-6. Oligos used to replace rplQ CUU 160-162

Oligo	Sequence
rplQ_CTT162YTR*	g*t*acgggcgaatgccagacgacgattagcaacgctatcagtcttggc**YAR**agtaatcagcggctcaactacgcggcgcagctctttcgct
rplQ_CTT162ATY*	g*t*acgggcgaatgccagacgacgattagcaacgctatcagtcttggc**RAT**agtaatcagcggctcaactacgcggcgcagctctttcgct
rplQ_CTT162GTD*	g*t*acgggcgaatgccagacgacgattagcaacgctatcagtcttggc**HAC**agtaatcagcggctcaactacgcggcgcagctctttcgct
rplQ_CTT162GCD*	g*t*acgggcgaatgccagacgacgattagcaacgctatcagtcttggc**HGC**agtaatcagcggctcaactacgcggcgcagctctttcgct

Table S6-7. Successful rplQ CUU 160-162 replacements

Amino acid	Codon	Codon count
Leu (unchanged)	CUU	57
Leu (synonymous)	CUA	4
	CUG	0
	UUA	4
	UUG	0
Ile	AUC	5
	AUU	6
Val	GUU	8
	GUA	5
	GUG	0
Ala	GCU	2
	GCA	2
	GCG	3
	Total	**96**

Codons are color-coded for each amino acid. CUG, UUG, and GUG codons (red) were never observed to replace CUU at positions 160-162.

Table S6-8. Primers and oligonucleotides used in this study.

Oligo name	Purpose	Oligo sequence
gfp-rpmH	forward gene primer for generating rpmH recoded cassette	ggatatccaataaagccattga
gfp-rpmD	forward gene primer for generating rpmD recoded cassette	tcgctgccaagcgtggtaa
gfp-rpmC	forward gene primer for generating rpmC recoded cassette	gcagcgaaactgccga
gfp-rpsR	forward gene primer for generating rpsR recoded cassette	ggttttgcatgccgagcag
gfp-rpmB	forward gene primer for generating rpmB recoded cassette	gccaataccccatacgaag
gfp-rpsP	forward gene primer for generating rpsP recoded cassette	actccgttcctcgatgg
gfp-rpsQ	forward gene primer for generating rpsQ recoded cassette	cgcgatgtcgcacgcg
gfp-rpmA	forward gene primer for generating rpmA recoded cassette	gatgtgaaaattactggcatca
gfp-rpsS	forward gene primer for generating rpsS recoded cassette	tgataaattcatcgtacgtcgcc
gfp-rpsN	forward gene primer for generating rpsN recoded cassette	ctgctggctgcctttg
gfp-rplU	forward gene primer for generating rplU recoded cassette	atagcgcactctgaatcattgaaaa
gfp-rpsJ	forward gene primer for generating rpsJ recoded cassette	gtctgaggagtaatcattttcgtt
gfp-rplX	forward gene primer for generating rplX recoded cassette	gttcatgaaaattatctctctggc
gfp-rplW	forward gene primer for generating rplW recoded cassette	acaaagtcgtaatgactgctg
gfp-rplV	forward gene primer for generating rplV recoded cassette	ggccacgctgctgataaa
gfp-rplS	forward gene primer for generating rplS recoded cassette	atggcgtaagcccccg
gfp-rplR	forward gene primer for generating rplR recoded cassette	ccgacgaagtcgtgcgta
gfp-rplT	forward gene primer for generating rplT recoded cassette	cgttaacgtttttaacttttttaattagaatataga
gfp-rpsM	forward gene primer for generating rpsM recoded cassette	aaacgggcttttcagca
gfp-rplL	forward gene primer for generating rplL recoded cassette	aacgcattcgcttacgtataaa
gfp-rplN	forward gene primer for generating rplN recoded cassette	cgacctgattttcgggtctc
gfp-rpsL	forward gene primer for generating rpsL recoded cassette	acgttttattacgtgtttacgaag
gfp-rplQ	forward gene primer for generating rplQ recoded cassette	tgacgagtaaccggatcac
gfp-rpsK	forward gene primer for generating rpsK recoded cassette	ccgtaagggtccgcgc
gfp-rpsI	forward gene primer for generating rpsI recoded cassette	acgcggcacagcaacc
gfp-rpsH	forward gene primer for generating rpsH recoded cassette	aaaaggctagctggtaattgt

Table S6-8 (Continued).		
gfp-rplM	forward gene primer for generating rplM recoded cassette	agacgtttgggtgttca
gfp-rplP	forward gene primer for generating rplP recoded cassette	ctcagcctaaaaagcagca
gfp-rplO	forward gene primer for generating rplO recoded cassette	gcggtatgatcaacgcgg
gfp-rplJ	forward gene primer for generating rplJ recoded cassette	tgaagtgagttccagagatttt
gfp-rpsE	forward gene primer for generating rpsE recoded cassette	gcagatgctgcccgtg
gfp-rplF	forward gene primer for generating rplF recoded cassette	ctggtcttggtggcgaa
gfp-rplE	forward gene primer for generating rplE recoded cassette	cgtttcttcaagtctaacagcg
gfp-rpsG	forward gene primer for generating rpsG recoded cassette	ctaaactcgtagagttttggacaa
gfp-rpsD	forward gene primer for generating rpsD recoded cassette	aaaaacgtcgcgtataacgcc
gfp-rplD	forward gene primer for generating rplD recoded cassette	cggtagcgacctgatcgt
gfp-rplC	forward gene primer for generating rplC recoded cassette	cagatcagcctgggtt
gfp-rpsC	forward gene primer for generating rpsC recoded cassette	agccacatcactgtggttg
gfp-rpsB	forward gene primer for generating rpsB recoded cassette	tatgggatacgtggaggca
gfp-rplB	forward gene primer for generating rplB recoded cassette	cagaatctggacttcgttgg
gfp-prfB	forward gene primer for generating prfB recoded cassette	tcccgctcttatcaccg
gfp-rpsA	forward gene primer for generating rpsA recoded cassette	catccggcatggagcc
grp-rpmH	reverse gene primer for generating rpmH recoded cassette	gacgtctttctagattattttgacacc
grp-rpmD	reverse gene primer for generating rpmD recoded cassette	gacgtctttctagattattcttccac
grp-rpmC	reverse gene primer for generating rpmC recoded cassette	gacgtctttctagattaagcg
grp-rpsR	reverse gene primer for generating rpsR recoded cassette	gacgtctttctagattattggtga
grp-rpmB	reverse gene primer for generating rpmB recoded cassette	gacgtctttctagattaatatttctcacc
grp-rpsP	reverse gene primer for generating rpsP recoded cassette	gacgtctttctagattatgcag
grp-rpsQ	reverse gene primer for generating rpsQ recoded cassette	gacgtctttctagattacaacaca
grp-rpmA	reverse gene primer for generating rpmA recoded cassette	gacgtctttctagattactccg
grp-rpsS	reverse gene primer for generating rpsS recoded cassette	gacgtctttctagattacttttttttg
grp-rpsN	reverse gene primer for generating rpsN recoded cassette	gacgtctttctagattaccacga
grp-rplU	reverse gene primer for generating rplU recoded cassette	gacgtctttctagattacgcc

Table S6-8 (Continued).		
grp-rpsJ	reverse gene primer for generating rpsJ recoded cassette	gacgtctttctagattagccca
grp-rplX	reverse gene primer for generating rplX recoded cassette	gacgtctttctagattatttaatcgtttc
grp-rplW	reverse gene primer for generating rplW recoded cassette	gacgtctttctagattattctgcac
grp-rplV	reverse gene primer for generating rplV recoded cassette	gacgtctttctagattatcggtc
grp-rplS	reverse gene primer for generating rplS recoded cassette	gacgtctttctagattaattcaggc
grp-rplR	reverse gene primer for generating rplR recoded cassette	gacgtctttctagattaaaattgcag
grp-rplT	reverse gene primer for generating rplT recoded cassette	gacgtctttctagattacgctaac
grp-rpsM	reverse gene primer for generating rpsM recoded cassette	gacgtctttctagattacttttttaataggc
grp-rplL	reverse gene primer for generating rplL recoded cassette	gacgtctttctagattacttcacc
grp-rplN	reverse gene primer for generating rplN recoded cassette	gacgtctttctagattataacacct
grp-rpsL	reverse gene primer for generating rpsL recoded cassette	gacgtctttctagattacgcttt
grp-rplQ	reverse gene primer for generating rplQ recoded cassette	gacgtctttctagattattcagcc
grp-rpsK	reverse gene primer for generating rpsK recoded cassette	gacgtctttctagattacactcgg
grp-rpsI	reverse gene primer for generating rpsI recoded cassette	gacgtctttctagattagcg
grp-rpsH	reverse gene primer for generating rpsH recoded cassette	gacgtctttctagattacgcaac
grp-rplM	reverse gene primer for generating rplM recoded cassette	gacgtctttctagattaaatatccag
grp-rplP	reverse gene primer for generating rplP recoded cassette	gacgtctttctagattacataacag
grp-rplO	reverse gene primer for generating rplO recoded cassette	gacgtctttctagattactcctca
grp-rplJ	reverse gene primer for generating rplJ recoded cassette	gacgtctttctagattacgctg
grp-rpsE	reverse gene primer for generating rpsE recoded cassette	gacgtctttctagattacttacct
grp-rplF	reverse gene primer for generating rplF recoded cassette	gacgtctttctagattatttttttttcg
grp-rplE	reverse gene primer for generating rplE recoded cassette	gacgtctttctagattatttacgaaatgg
grp-rpsG	reverse gene primer for generating rpsG recoded cassette	gacgtctttctagattagttcagataa
grp-rpsD	reverse gene primer for generating rpsD recoded cassette	gacgtctttctagattattttgaatac
grp-rplD	reverse gene primer for generating rplD recoded cassette	gacgtctttctagattacgcc
grp-rplC	reverse gene primer for generating rplC recoded cassette	gacgtctttctagattatgctttaac
grp-rpsC	reverse gene primer for generating rpsC recoded cassette	gacgtctttctagattactttcgt

Table S6-8 (Continued).		
grp-rpsB	reverse gene primer for generating rpsB recoded cassette	gacgtctttctagattattccgc
grp-rplB	reverse gene primer for generating rplB recoded cassette	gacgtctttctagattacttcgat
grp-prfB	reverse gene primer for generating prfB recoded cassette	gacgtctttctagattataggc
grp-rpsA	reverse gene primer for generating rpsA recoded cassette	tgacgtctttctagattattcacc
kfp-rpmH	forward kanR primer for generating rpmH recoded cassette	gtgcgcgcttaacggtgtcaaaataatctagaaagacgttgagttgtcg agattttcagg
kfp-rpmD	forward kanR primer for generating rpmD recoded cassette	cgtttatggtgaaggtggaagaataatctagaaagacgttgagttgtcg agattttcagg
kfp-rpmC	forward kanR primer for generating rpmC recoded cassette	tgttgaatgaaaaagctggcgcttaatctagaaagacgttgagttgtcga gattttcagg
kfp-rpsR	forward kanR primer for generating rpsR recoded cassette	taccttatacggaccgtcaccaataatctagaaagacgttgagttgtcga gattttcagg
kfp-rpmB	forward kanR primer for generating rpmB recoded cassette	tacgcgcgcgcggtgagaaatattaatctagaaagacgttgagttgtcg agattttcagg
kfp-rpsP	forward kanR primer for generating rpsP recoded cassette	ttaaggaggtgaataaggctgcataatctagaaagacgttgagttgtcg agattttcagg
kfp-rpsQ	forward kanR primer for generating rpsQ recoded cassette	gtgtggtggaaaaggctgtgttgtaatctagaaagacgttgagttgtcg agattttcagg
kfp-rpmA	forward kanR primer for generating rpmA recoded cassette	agttcatttcgattgaggcggagtaatctagaaagacgttgagttgtcga gattttcagg
kfp-rpsS	forward kanR primer for generating rpsS recoded cassette	cagacaagaaggcaaaaaaaaagtaatctagaaagacgttgagttgtc gagattttcagg
kfp-rpsN	forward kanR primer for generating rpsN recoded cassette	caggcctaaagaaagcatcgtggtaatctagaaagacgttgagttgtc gagattttcagg
kfp-rplU	forward kanR primer for generating rplU recoded cassette	taaagatcacaggtatttcggcgtaatctagaaagacgttgagttgtcga gattttcagg
kfp-rpsJ	forward kanR primer for generating rpsJ recoded cassette	tggatgttcaaattagtttgggctaatctagaaagacgttgagttgtcgag attttcagg
kfp-rplX	forward kanR primer for generating rplX recoded cassette	aaagcaattcggaaacgattaaataatctagaaagacgttgagttgtcg agattttcagg
kfp-rplW	forward kanR primer for generating rplW recoded cassette	tggattttgtaggaggtgcagaataatctagaaagacgttgagttgtcga gattttcagg
kfp-rplV	forward kanR primer for generating rplV recoded cassette	ttacggttgtggtaagcgaccgataatctagaaagacgttgagttgtcg agattttcagg
kfp-rplS	forward kanR primer for generating rplS recoded cassette	cgcgaattaaggaacgcctgaattaatctagaaagacgttgagttgtcg agattttcagg
kfp-rplR	forward kanR primer for generating rplR recoded cassette	cacgcgaggcgggactgcaattttaatctagaaagacgttgagttgtcg agattttcagg
kfp-rplT	forward kanR primer for generating rplT recoded cassette	agaaggctaaggctgcgttagcgtaatctagaaagacgttgagttgtcg agattttcagg
kfp-rpsM	forward kanR primer for generating rpsM recoded cassette	gcccacgtaagcctattaaaaagtaatctagaaagacgttgagttgtcg agattttcagg
kfp-rplL	forward kanR primer for generating rplL recoded cassette	caggtgcagaggtagaggtgaagtaatctagaaagacgttgagttgtc gagattttcagg
kfp-rplN	forward kanR primer for generating rplN recoded cassette	ttagcttggcgccggaggtgttataatctagaaagacgttgagttgtcga gattttcagg
kfp-rpsL	forward kanR primer for generating rpsL recoded cassette	acggtgttaaacgaccgaaagcgtaatctagaaagacgttgagttgtc gagattttcagg
kfp-rplQ	forward kanR primer for generating rplQ recoded cassette	aaaaggctgaggcagcggctgaataatctagaaagacgttgagttgtc gagattttcagg

Table S6-8 (Continued).		
kfp-rpsK	forward kanR primer for generating rpsK recoded cassette	gccctccaaagaagcgccgagtgtaatctagaaagacgttgagttgtcgagattttcagg
kfp-rpsI	forward kanR primer for generating rpsI recoded cassette	gccgacctcaatttagtaagcgctaatctagaaagacgttgagttgtcgagattttcagg
kfp-rpsH	forward kanR primer for generating rpsH recoded cassette	gtgagatcatttgttatgttgcgtaatctagaaagacgttgagttgtcgagattttcagg
kfp-rplM	forward kanR primer for generating rplM recoded cassette	aacagcctcaggtgctggatatttaatctagaaagacgttgagttgtcgagattttcagg
kfp-rplP	forward kanR primer for generating rplP recoded cassette	cattcgtgacgaaaactgttatgtaatctagaaagacgttgagttgtcgagattttcagg
kfp-rplO	forward kanR primer for generating rplO recoded cassette	cggcgggtggcaagattgaggagtaatctagaaagacgttgagttgtcgagattttcagg
kfp-rplJ	forward kanR primer for generating rplJ recoded cassette	tgcgtgacgctaaggaggcagcgtaatctagaaagacgttgagttgtcgagattttcagg
kfp-rpsE	forward kanR primer for generating rpsE recoded cassette	gcgtggaggagatcctaggtaagtaatctagaaagacgttgagttgtcgagattttcagg
kfp-rplF	forward kanR primer for generating rplF recoded cassette	gcactaaggaagcgaaaaaaaaataatctagaaagacgttgagttgtcgagattttcagg
kfp-rplE	forward kanR primer for generating rplE recoded cassette	cgttcgattttccatttcgtaaataatctagaaagacgttgagttgtcgagattttcagg
kfp-rpsG	forward kanR primer for generating rpsG recoded cassette	aaccggcgttaggttatctgaactaatctagaaagacgttgagttgtcgagattttcagg
kfp-rpsD	forward kanR primer for generating rpsD recoded cassette	taattgtggaattgtattcaaaataatctagaaagacgttgagttgtcgagattttcagg
kfp-rplD	forward kanR primer for generating rplD recoded cassette	aacaggtagaagaaatgttggcgtaatctagaaagacgttgagttgtcgagattttcagg
kfp-rplC	forward kanR primer for generating rplC recoded cassette	ttgtgaagccggcggttaaagcataatctagaaagacgttgagttgtcgagattttcagg
kfp-rpsC	forward kanR primer for generating rpsC recoded cassette	aacaacaacgcaagggacgaaagtaatctagaaagacgttgagttgtcgagattttcagg
kfp-rpsB	forward kanR primer for generating rpsB recoded cassette	aggagtcttttgttgaggcggaataatctagaaagacgttgagttgtcgagattttcagg
kfp-rplB	forward kanR primer for generating rplB recoded cassette	ttattgtgcgccgtcgatcgaagtaatctagaaagacgttgagttgtcgagattttcagg
kfp-prfB	forward kanR primer for generating prfB recoded cassette	aggcgtctttaaaggcgggcctataatctagaaagacgttgagttgtcgagattttcagg
kfp-rpsA	forward kanR primer for generating rpsA recoded cassette	atttaaggcggcgaagggtgaataatctagaaagacgtctgagttgtcgagattttcagg
krp-rpmH	reverse kanR primer for generating rpmH recoded cassettes and NAT_*kan*^{R}	agcgtaactccctgggaaatgcgagcttaaccactcaggggttagctttattagaaaaactcatcgagcatc
krp-rpmD	reverse kanR primer for generating rpmD recoded cassettes and NAT_*kan*^{R}	cccgccttttggagccttcggccggagacagagtatttaaacgcatcttcttagaaaaactcatcgagcatc
krp-rpmC	reverse kanR primer for generating rpmC recoded cassettes and NAT_*kan*^{R}	cattttgtcgctaacaacgcgaccttgcagagtacggattttatcggtcattacgcacccgccttcttagaaaaactcatcgagcatc
krp-rpsR	reverse kanR primer for generating rpsR recoded cassettes and NAT_*kan*^{R}	aacttgcattaccttatcctctcaaagtcgtattaatggaccgtgaccgatatagaaaaactcatcgagcatc
krp-rpmB	reverse kanR primer for generating rpmB recoded cassettes and NAT_*kan*^{R}	cttgattttctcacgaataccttagccatgatttatttcctctaagtacttagaaaaactcatcgagcatc
krp-rpsP	reverse kanR primer for generating rpsP recoded cassettes and NAT_*kan*^{R}	acaggtgcttgcgcggtgagttgtttgctcatcatgaccaccgtgacagattagaaaaactcatcgagcatc
krp-rpsQ	reverse kanR primer for generating rpsQ recoded cassettes and NAT_*kan*^{R}	taaacggctcatttctgagccgtttattcgtattgagagagtgtactgtattagaaaaactcatcgagcatc
krp-rpmA	reverse kanR primer for generating rpmA recoded cassettes and NAT_*kan*^{R}	gccccgcaacgtgttgcggggctttcatccgttaccgggacgcgaaaaacttagaaaaactcatcgagcatc

Table S6-8 (Continued).		
krp-rpsS	reverse kanR primer for generating rpsS recoded cassettes and NAT_*kan*^{R}	agaacgagcatggcgatgtttagcgatagtttccatctcttcctcctacct tagaaaaactcatcgagcatc
krp-rpsN	reverse kanR primer for generating rpsN recoded cassettes and NAT_*kan*^{R}	ggatcttgcatgctcatctgtctttactcccgtgattcaattggtgacaatt agaaaaactcatcgagcatc
krp-rplU	reverse kanR primer for generating rplU recoded cassettes and NAT_*kan*^{R}	tgtggagccgccagccttttatgtgccatttgaaatctctcctcaggtctt agaaaaactcatcgagcatc
krp-rpsJ	reverse kanR primer for generating rpsJ recoded cassettes and NAT_*kan*^{R}	accgactaaaccaatcattgtttcaacctctcaatcgctcaatgacctgat tagaaaaactcatcgagcatc
krp-rplX	reverse kanR primer for generating rplX recoded cassettes and NAT_*kan*^{R}	tacttcgtctttgtagtaatcatgcagtttcgccatcgtactactccaaatta gaaaaactcatcgagcatc
krp-rplW	reverse kanR primer for generating rplW recoded cassettes and NAT_*kan*^{R}	cggagatgtcggtttacatttaacaactgccattgtattactcctccgactt agaaaaactcatcgagcatc
krp-rplV	reverse kanR primer for generating rplV recoded cassettes and NAT_*kan*^{R}	gcgaataccattaggatgtactttctgacccattgctagtctccagagtct tagaaaaactcatcgagcatc
krp-rplS	reverse kanR primer for generating rplS recoded cassettes and NAT_*kan*^{R}	gccagccaattggccagcccttcttaacaggatgtcgcttaagcgaaat cttagaaaaactcatcgagcatc
krp-rplR	reverse kanR primer for generating rplR recoded cassettes and NAT_*kan*^{R}	ctgcagttcgccagcttgtttttcgatgtgagccatcttacacctctacctt agaaaaactcatcgagcatc
krp-rplT	reverse kanR primer for generating rplT recoded cassettes and NAT_*kan*^{R}	tgatggcgttgaaacgaaaagagggagactagctccctctttcaactg gcttagaaaaactcatcgagcatc
krp-rpsM	reverse kanR primer for generating rpsM recoded cassettes and NAT_*kan*^{R}	cacgtttacgtgcacgaattggtgcctttgccattattcaatcaccccgat tagaaaaactcatcgagcatc
krp-rplL	reverse kanR primer for generating rplL recoded cassettes and NAT_*kan*^{R}	agtcaccagccatcagcctgatttctcaggctgcaaccggaagggttg gcttagaaaaactcatcgagcatc
krp-rplN	reverse kanR primer for generating rplN recoded cassettes and NAT_*kan*^{R}	acacgataacttcgtcatcacgacggattttcgctgccatgattcgctcct tagaaaaactcatcgagcatc
krp-rpsL	reverse kanR primer for generating rpsL recoded cassettes and NAT_*kan*^{R}	ttagtttgacatttaagttaaaacgtttggccttacttaacggagaaccatt agaaaaactcatcgagcatc
krp-rplQ	reverse kanR primer for generating rplQ recoded cassettes and NAT_*kan*^{R}	tacgggtataaaaaaacccgccggggcgggttttttacgttgcttcaga ttagaaaaactcatcgagcatc
krp-rpsK	reverse kanR primer for generating rpsK recoded cassettes and NAT_*kan*^{R}	cccaaatatcttgccattttctttctccaacaaacctggaaaacgaggcgt tagaaaaactcatcgagcatc
krp-rpsI	reverse kanR primer for generating rpsI recoded cassettes and NAT_*kan*^{R}	cgccgaagcgggtttttcgaaaattgttttctgccggagcagaagcca attagaaaaactcatcgagcatc
krp-rpsH	reverse kanR primer for generating rpsH recoded cassettes and NAT_*kan*^{R}	gcaggaacaacgaccggtgctttagcaacacgagacatttttttcctccg attagaaaaactcatcgagcatc
krp-rplM	reverse kanR primer for generating rplM recoded cassettes and NAT_*kan*^{R}	cggcgaccagtgccgtagtattgattttcagccattgcctataatcccga ttagaaaaactcatcgagcatc
krp-rplP	reverse kanR primer for generating rplP recoded cassettes and NAT_*kan*^{R}	ctcggtgttcagctcttcaacgctcttctcacgcagctcttttgctttcatta catcaccgtcttattagaaaaactcatcgagcatc
krp-rplO	reverse kanR primer for generating rplO recoded cassettes and NAT_*kan*^{R}	caccttggcactttgaaaatctaatcccggttgtttagccatctgctactt agaaaaactcatcgagcatc
krp-rplJ	reverse kanR primer for generating rplJ recoded cassettes and NAT_*kan*^{R}	tatcagaataagtttatacgtaagcgaatgcgttaaaaagataactgcga ttagaaaaactcatcgagcatc
krp-rpsE	reverse kanR primer for generating rpsE recoded cassettes and NAT_*kan*^{R}	cgaccgattgcactgcgggtttgagtaattttaatagtctttgccatggttt agaaaaactcatcgagcatc
krp-rplF	reverse kanR primer for generating rplF recoded cassettes and NAT_*kan*^{R}	gcgcgggtcgcacgacggatacgagcagatttcttatccatagtgttac cttagaaaaactcatcgagcatc
krp-rplE	reverse kanR primer for generating rplE recoded cassettes and NAT_*kan*^{R}	ttttacttcgcgtgctttcattgattgcttagccatttagtaaccctaccttag aaaaactcatcgagcatc
krp-rpsG	reverse kanR primer for generating rpsG recoded cassettes and NAT_*kan*^{R}	gcgatgggtgttgtacgagccatttgtttcctcgtttatctttttaggcgttta gaaaaactcatcgagcatc
krp-rpsD	reverse kanR primer for generating rpsD recoded cassettes and NAT_*kan*^{R}	gaaactctgtcacagaaccctgcattgtgtcctctctttggtactaagcttt agaaaaactcatcgagcatc

Table S6-8 (Continued).		
krp-rplD	reverse kanR primer for generating rplD recoded cassettes and NAT_*kan*^{R}	tcagaaacgtgcggtgcacgcagcaccttcagcagacgttcttcacgaatcatgccagcatctcctcattagaaaaactcatcgagcatc
krp-rplC	reverse kanR primer for generating rplC recoded cassettes and NAT_*kan*^{R}	cagtcagcgcgctctgcgcgtctttcaatactaattccattgctatctccttagaaaaactcatcgagcatc
krp-rpsC	reverse kanR primer for generating rpsC recoded cassettes and NAT_*kan*^{R}	tgcattttacggaattttgtacgctttggttgtaacatcagcgacgctccttagaaaaactcatcgagcatc
krp-rpsB	reverse kanR primer for generating rpsB recoded cassettes and NAT_*kan*^{R}	ttgccgcctttctgcaactcgaactattttgggggagttatcaagccttattagaaaaactcatcgagcatc
krp-rplB	reverse kanR primer for generating rplB recoded cassettes and NAT_*kan*^{R}	caataaaaggacctttcttgagagaacgtggcatggcttatcctctaaaattagaaaaactcatcgagcatc
krp-prfB	reverse kanR primer for generating prfB recoded cassettes and NAT_*kan*^{R}	tcgactaccgcgtcagcgccctgtgcgtgttgttcagacatgttggttccttagaaaaactcatcgagcatc
krp-rpsA	reverse kanR primer for generating rpsA recoded cassettes and NAT_*kan*^{R}	tcaagtaaactcaacaaacttcggaataaaaatcccgaagagtcagagaattagaaaaactcatcgagcatc
prfB-1r	prfB split synthesis N-terminus reverse primer	cacccgaaatcttaatagtaacgct
prfB-2f	prfB split synthesis C-terminus forward primer	gacggagattattgaggaatctgag
rpsA-1r	rpsA split synthesis N-terminus reverse primer	gtaacacgccctgttaacttcg
rpsA-2f	rpsA split synthesis C-terminus forward primer	gcaattgcgaagcgctac
rpmH-early	C-terminal cassette forward primer for rpmH; All but 90 bp is recoded	gttctcacggcttccgtgctcgtatggctactaaaaatggtcgtcaggtttttggcgcgccgccgc
rpsR-early	C-terminal cassette forward primer for rpsR; All but 90 bp is recoded	ccgcggaaggcgttcaagagatcgactataaagatatcgctacgctgaaaaattatattacggagtctggcaaaa
rpsP-early	C-terminal cassette forward primer for rpsP; All but 90 bp is recoded	gtccgttctaccaggttgttgtcgctgacagccgtaatgcacgcaacggtcgttttattgaacgtgtgggc
rplT-early	C-terminal cassette forward primer for rplT; All but 90 bp is recoded	acaagaaaattttgaaacaagctaaaggctactacggtgcgcgttctcgcgtgtatcgtgtagcttttcaagc
rplL-early	C-terminal cassette forward primer for rplL; All but 90 bp is recoded	ctatgtctgtaatggacgttgtagaactgatctctgcaatggaagaaaaatttggcgtatcagcagcg
rpsL-early	C-terminal cassette forward primer for rpsL; All but 90 bp is recoded	gcaaagttgcgaaaagcaacgtgcctgcgctggaagcatgcccgcaaaaacgcggggtttgcacg
rplQ-early	C-terminal cassette forward primer for rplQ; All but 90 bp is recoded	gcagccatcgccaggctatgttccgcaatatggcaggttcactggttcgtcacgagattattaaaactacattaccga
rplO-early	C-terminal cassette forward primer for rplO; All but 90 bp is recoded	aggcgggtaaacgcctgggtcgtggtatcggttctggcctcggtaaaaccggcggacgcggccat
rplJ-early	C-terminal cassette forward primer for rplJ; All but 90 bp is recoded	aagtcagcgaagtagccaaaggcgcgctgtctgcagtagttgcggattcccgcggtgtgacagtgg
rpsG-early	C-terminal cassette forward primer for rpsG; All but 90 bp is recoded	cggatccgaagttcggatcagaactgctggctaaatttgtaaatatcctgatggtggacgggaaaaaga
rpsD-early	C-terminal cassette forward primer for rpsD; All but 90 bp is recoded	gtgagggcaccgacttattccttaagtctggcgttcgcgcgatcgataccaaatgcaagatcgagcaggc
rplD-early	C-terminal cassette forward primer for rplD; All but 90 bp is recoded	tttccgaaactaccttcggtcgtgatttcaacgaagcgctggttcaccaggtagtggtggcgtacgc
rpsC-early	C-terminal cassette forward primer for rpsC; All but 90 bp is recoded	ttgtaaaaccatggaactctacctggtttgcgaacaccaaagaattcgctgataatttagatagtgacttcaaggttcg
rpsB-early	C-terminal cassette forward primer for rpsB; All but 90 bp is recoded	ttcacttcggtcaccagacccgttactggaacccgaaaatgaagccgttcatttttggcgcacgcaa
rplB-early	C-terminal cassette forward primer for rplB; All but 90 bp is recoded	gccacgtagttaaagtggttaaccctgagctgcacaagggcaaaccttttgcgccattattagagaagaattct
rpsA-early	C-terminal cassette forward primer for rpsA; All but 90 bp is recoded	aagaaatcgaaacccgcccgggttctatcgttcgtggcgttgttgttgctattgataaggatgttgtgttggtg

Table S6-8 (Continued).		
rpmH-middle	C-terminal cassette forward primer for rpmH; Half of gene is recoded	cgtctgtactgaagcgcaaccgttctcacggcttccgtgctcgtatggctacgaagaacggccgc
rpsR-middle	C-terminal cassette forward primer for rpsR; Half of gene is recoded	actataaagatatcgctacgctgaaaaactacatcaccgaaagcggtaagatcgttccttcacgcattaca
rpsP-middle	C-terminal cassette forward primer for rpsP; Half of gene is recoded	gtaatgcacgcaacggtcgcttcatcgagcgcgttggtttcttcaacccaattgcgtctgagaaggagg
rplT-middle	C-terminal cassette forward primer for rplT; Half of gene is recoded	gtcagtatgcttaccgtgaccgtcgtcaacgtaagcgtcagttccgtcaattatggatcgcacgcattaatg
rplL-middle	C-terminal cassette forward primer for rplL; Half of gene is recoded	cggttgaagctgctgaagaaaaaactgaattcgacgtaattctgaaagctgcgggggcgaataagg
rpsL-middle	C-terminal cassette forward primer for rpsL; Half of gene is recoded	actccgcgctgcgtaaagtatgccgtgttcgtctgactaacggtttcgaagttacgtcttatattggcggag
rplQ-middle	C-terminal cassette forward primer for rplQ; Half of gene is recoded	ttgagccgctgattactcttgccaagactgatagcgttgctaatcgtcgttggcgtttgctcgca
rplO-middle	C-terminal cassette forward primer for rplO; Half of gene is recoded	ctctgtaccgtcgtctgccgaaattcggcttcacttctcgtaaagcagcgattactgcggagatccgc
rplJ-middle	C-terminal cassette forward primer for rplJ; Half of gene is recoded	ctccgttcgagtgcctgaaagacgcgtttgttggtccgaccctgattgcatatagcatggagcatcctgg
rpsG-middle	C-terminal cassette forward primer for rpsG; Half of gene is recoded	aagttaagtctcgccgcgttggtggttctacttatcaggtaccagttgaagtgcgacctgtacgcc
rpsD-middle	C-terminal cassette forward primer for rpsD; Half of gene is recoded	gtgaaaacctgttggctctgctggaaggtcgtctggacaacgttgtataccgcatggggtttggcg
rplD-middle	C-terminal cassette forward primer for rplD; Half of gene is recoded	ttgctgctcgtccgcaggaccacagtcaaaaagttaacaagaagatgtaccgaggggcattaaagtctatttt
rpsC-middle	C-terminal cassette forward primer for rpsC; Half of gene is recoded	tcaacatcgccgaagttcgtaagcctgaactggacgcaaaactggttgctgatagtattacaagccaattagagcg
rpsB-middle	C-terminal cassette forward primer for rpsB; Half of gene is recoded	aaaccgttcgtcagtccatcaaacgtctgaaagacctggaaactcagtctcaagatggcacgtttgataaattaa
rplB-middle	C-terminal cassette forward primer for rplB; Half of gene is recoded	atgctgcaatcaaaccaggtaacaccctgccgatgcgcaacatcccggttggaagcacggtgcaca
rpsA-middle	C-terminal cassette forward primer for rpsA; Half of gene is recoded	gcgaagatccgtgggtagctatcgctaaacgttatccggaaggtaccaaattaacagggcgtgttactaatttg
NAT_krp-rpmH	forward kanR primer for generating rpmH NAT_*kan*^{R} cassette	tcgtgctaaaggccgcgctcgtctgaccgtttctaagtaatgagttgtcgagattttcagg
NAT_krp-rpmD	forward kanR primer for generating rpmD NAT_*kan*^{R} cassette	gatcaacgcggtttccttcatggttaaagttgaggagtaatgagttgtcgagattttcagg
NAT_krp-rpmC	forward kanR primer for generating rpmC NAT_*kan*^{R} cassette	acgcgttaagactttactgaacgagaaggcgggtgcgtaatgagttgtcgagattttcagg
NAT_krp-rpsR	forward kanR primer for generating rpsR NAT_*kan*^{R} cassette	ctacctgtccctgctgccgtacactgatcgccatcagtaatgagttgtcgagattttcagg
NAT_krp-rpmB	forward kanR primer for generating rpmB NAT_*kan*^{R} cassette	agttctggctgaactgcgtgcccgtggcgaaaagtactaatgagttgtcgagattttcagg
NAT_krp-rpsP	forward kanR primer for generating rpsP NAT_*kan*^{R} cassette	cgttgctgcgctgatcaaagaagtaaacaaagcagcttaatgagttgtcgagattttcagg
NAT_krp-rpsQ	forward kanR primer for generating rpsQ NAT_*kan*^{R} cassette	ctggacgctggttcgcgttgtagagaaagcggttctgtaatgagttgtcgagattttcagg
NAT_krp-rpmA	forward kanR primer for generating rpmA NAT_*kan*^{R} cassette	cccgaaaaaccgtaaatttatcagcatcgaagctgaataatgagttgtcgagattttcagg
NAT_krp-rpsS	forward kanR primer for generating rpsS NAT_*kan*^{R} cassette	tcgcggccacgctgctgataaaaaagcgaagaagaaataatgagttgtcgagattttcagg
NAT_krp-rpsN	forward kanR primer for generating rpsN NAT_*kan*^{R} cassette	gcgcggtgaaatcccgggtctgaaaaaggctagctggtaatgagttgtcgagattttcagg
NAT_krp-rplU	forward kanR primer for generating rplU NAT_*kan*^{R} cassette	gtggttcactgatgtgaaaattactggcatcagcgcctaatgagttgtcgagattttcagg

Table S6-8 (Continued).		
NAT_krp-rpsJ	forward kanR primer for generating rpsJ NAT_*kan*^{R} cassette	tctggctgccggtgtagacgtgcagatcagcctgggttaatgagttgtc gagattttcagg
NAT_krp-rplX	forward kanR primer for generating rplX NAT_*kan*^{R} cassette	agtccgtttcttcaagtctaacagcgaaactatcaagtaatgagttgtcg agattttcagg
NAT_krp-rplW	forward kanR primer for generating rplW NAT_*kan*^{R} cassette	agaaggccagaatctggacttcgttggcggcgctgagtaatgagttgt cgagattttcagg
NAT_krp-rplV	forward kanR primer for generating rplV NAT_*kan*^{R} cassette	gcgcaccagccacatcactgtggttgtgtccgatcgctgatgagttgtc gagattttcagg
NAT_krp-rplS	forward kanR primer for generating rplS NAT_*kan*^{R} cassette	tactggtaaggctgctcgtatcaaagagcgtcttaactaatgagttgtcg agattttcagg
NAT_krp-rplR	forward kanR primer for generating rplR NAT_*kan*^{R} cassette	actggcagatgctgcccgtgaagctggccttcagttctaatgagttgtc gagattttcagg
NAT_krp-rplT	forward kanR primer for generating rplT NAT_*kan*^{R} cassette	caccgctctggttgaaaaagcgaaagcagctctggcataatgagttgtc gagattttcagg
NAT_krp-rpsM	forward kanR primer for generating rpsM NAT_*kan*^{R} cassette	acgtacccgtaagggtccgcgcaaaccgatcaagaaataatgagttgt cgagattttcagg
NAT_krp-rplL	forward kanR primer for generating rplL NAT_*kan*^{R} cassette	agctctggaagaagctggcgctgaagttgaagttaaataatgagttgtc gagattttcagg
NAT_krp-rplN	forward kanR primer for generating rplN NAT_*kan*^{R} cassette	gttcatgaaaattatctctctggcaccagaagtactctaatgagttgtcga gattttcagg
NAT_krp-rpsL	forward kanR primer for generating rpsL NAT_*kan*^{R} cassette	ggctcgttccaagtatggcgtgaagcgtcctaaggcttaatgagttgtc gagattttcagg
NAT_krp-rplQ	forward kanR primer for generating rplQ NAT_*kan*^{R} cassette	ggttgatcgttcagagaaagcagaagctgctgcagagtaatgagttgtc gagattttcagg
NAT_krp-rpsK	forward kanR primer for generating rpsK NAT_*kan*^{R} cassette	tcataacggttgtcgtccgccgaaaaaacgtcgcgtataatgagttgtc gagattttcagg
NAT_krp-rpsI	forward kanR primer for generating rpsI NAT_*kan*^{R} cassette	gcgtaaagcacgtcgtcgtccgcagttctccaaacgttaatgagttgtc gagattttcagg
NAT_krp-rpsH	forward kanR primer for generating rpsH NAT_*kan*^{R} cassette	ggctggtcttggtggcgaaattatctgctacgtagcctaatgagttgtcg agattttcagg
NAT_krp-rplM	forward kanR primer for generating rplM NAT_*kan*^{R} cassette	caaccacgcggcacagcaaccgcaagttcttgacatctaatgagttgtc gagattttcagg
NAT_krp-rplP	forward kanR primer for generating rplP NAT_*kan*^{R} cassette	gccgattaaaaccacctttgtaactaagacggtgatgtaatgagttgtcg agattttcagg
NAT_krp-rplO	forward kanR primer for generating rplO NAT_*kan*^{R} cassette	tgctgctatcgaagctgctggcggtaaaatcgaggaataatgagttgtc gagattttcagg
NAT_krp-rplJ	forward kanR primer for generating rplJ NAT_*kan*^{R} cassette	tactctggctgctgtacgcgatgcgaaagaagctgcttaatgagttgtc gagattttcagg
NAT_krp-rpsE	forward kanR primer for generating rpsE NAT_*kan*^{R} cassette	caagcgtggtaaatccgttgaagaaattctggggaaataatgagttgtc gagattttcagg
NAT_krp-rplF	forward kanR primer for generating rplF NAT_*kan*^{R} cassette	cgacgaagtcgtgcgtaccaaagaggctaagaagaagtaatgagttgt cgagattttcagg
NAT_krp-rplE	forward kanR primer for generating rplE NAT_*kan*^{R} cassette	cgctctgctggctgcctttgacttcccgttccgcaagtaatgagttgtcg agattttcagg
NAT_krp-rpsG	forward kanR primer for generating rpsG NAT_*kan*^{R} cassette	cgcttccagtaagcagcccgctttgggctacttaaattgatgagttgtcg agattttcagg
NAT_krp-rpsD	forward kanR primer for generating rpsD NAT_*kan*^{R} cassette	cattaacgaacacctgatcgtcgagctttactccaagtaatgagttgtcg agattttcagg
NAT_krp-rplD	forward kanR primer for generating rplD NAT_*kan*^{R} cassette	tgctgatgctgttaagcaagttgaggagatgctggcatgatgagttgtc gagattttcagg
NAT_krp-rplC	forward kanR primer for generating rplC NAT_*kan*^{R} cassette	cggtagcgacctgatcgttaaaccagctgtgaaggcgtaatgagttgtc gagattttcagg
NAT_krp-rpsC	forward kanR primer for generating rpsC NAT_*kan*^{R} cassette	tgctcagcctaaaaagcagcagcgtaaaggccgtaaataatgagttgt cgagattttcagg

Table S6-8 (Continued).		
NAT_krp-rpsB	forward kanR primer for generating rpsB NAT_*kan*^{R} cassette	ggcttcccaggcggaagaaagcttcgtagaagctgagtaatgagttgtcgagattttcagg
NAT_krp-rplB	forward kanR primer for generating rplB NAT_*kan*^{R} cassette	gcgtactgataaattcatcgtacgtcgccgtagcaaataatgagttgtcgagattttcagg
NAT_krp-prfB	forward kanR primer for generating prfB NAT_*kan*^{R} cassette	ggatcaatttatcgaagcaagtttgaaagcagggttatgatgagttgtcgagattttcagg
NAT_krp-rpsA	forward kanR primer for generating rpsA NAT_*kan*^{R} cassette	cgcaatggctgaagctttcaaagcagctaaaggcgagtaatgagttgtcgagattttcagg
bndfp-rpmH	rpmH forward boundary primer	tcggtgtccatcgtttca
bndfp-rpmD	rpmD forward boundary primer	ttgatggcctggaaaatatgaat
bndfp-rpmC	rpmC forward boundary primer	tgaagcattcaagctggc
bndfp-rpsR	rpsR forward boundary primer	caaagaacggactgagcaaa
bndfp-rpmB	rpmB forward boundary primer	gctgtaaagcctgacgag
bndfp-rpsP	rpsP forward boundary primer	ttcgggcttttaatatgacacc
bndfp-rpsQ	rpsQ forward boundary primer	gtctcacctgttgaagcaag
bndfp-rpmA	rpmA forward boundary primer	gccatcgtcagtggttca
bndfp-rpsS	rpsS forward boundary primer	taagaagacccgcagcaa
bndfp-rpsN	rpsN forward boundary primer	tgcgaaatctgacgaagaag
bndfp-rplU	rplU forward boundary primer	tattcgcgccctattgtga
bndfp-rpsJ	rpsJ forward boundary primer	cactctcccatcaatcgtaatg
bndfp-rplX	rplX forward boundary primer	ctcgtgagcttcgtagtga
bndfp-rplW	rplW forward boundary primer	ggttagcctgatcgcctt
bndfp-rplV	rplV forward boundary primer	aattcgcaccgactcgta
bndfp-rplS	rplS forward boundary primer	cgcacaacagcaacataaac
bndfp-rplR	rplR forward boundary primer	ccttataaaggcaagggtgttc
bndfp-rplT	rplT forward boundary primer	ctggtaatcgcgtgcctg
bndfp-rpsM	rpsM forward boundary primer	tctgtgcgtttccatttgag
bndfp-rplL	rplL forward boundary primer	gaagctgcttaatcgcagt
bndfp-rplN	rplN forward boundary primer	gccctcgatatggggatt
bndfp-rpsL	rpsL forward boundary primer	aaattcggcgtcctcatattg
bndfp-rplQ	rplQ forward boundary primer	catgcgcctggaaaactg
bndfp-rpsK	rpsK forward boundary primer	gtaccaagaccaacgcac
bndfp-rpsI	rpsI forward boundary primer	gtttacgcgggtaacgag
bndfp-rpsH	rpsH forward boundary primer	ctatgcgcggtgaaatcc
bndfp-rplM	rplM forward boundary primer	tttgtcgtgtgaacctcaac
bndfp-rplP	rplP forward boundary primer	ctgttgaacaaccggaaaaac
bndfp-rplO	rplO forward boundary primer	gcgaggatactcctgctatt
bndfp-rplJ	rplJ forward boundary primer	attaagacgctctctccgtt
bndfp-rpsE	rpsE forward boundary primer	caatatcatggtcgtgtccag
bndfp-rplF	rplF forward boundary primer	acctctaaaggtgttatgactga

Table S6-8 (Continued).		
bndfp-rplE	rplE forward boundary primer	ggctttagattcgaagacgg
bndfp-rpsG	rpsG forward boundary primer	ccgttaagtaaggccaaacg
bndfp-rpsD	rpsD forward boundary primer	ctcataacggttgtcgtcc
bndfp-rplD	rplD forward boundary primer	ctggttaaaggtgctgtcc
bndfp-rplC	rplC forward boundary primer	ctctgatgcgtctggatct
bndfp-rpsC	rpsC forward boundary primer	tgcagatcgcatcctgaa
bndfp-rpsB	rpsB forward boundary primer	cacatattccggggtgcc
bndfp-rplB	rplB forward boundary primer	agcttacgtcaccctgaaa
bndfp-prfB	prfB forward boundary primer	aaaaagagcgtggattggg
bndfp-rpsA	rpsA forward boundary primer	gaatgacagcgggtatgtt
bndrp-rpmH	rpmH reverse boundary primer	gaatgtgaattgactgggagtt
bndrp-rpmD	rpmD reverse boundary primer	gaaccgataccacgaccc
bndrp-rpmC	rpmC reverse boundary primer	gttcgatagcaacaacaatgga
bndrp-rpsR	rpsR reverse boundary primer	ctacccaggtttgctactttatc
bndrp-rpmB	rpmB reverse boundary primer	agtaccagcagaagaaacca
bndrp-rpsP	rpsP reverse boundary primer	catttttcccaaaacgatggg
bndrp-rpsQ	rpsQ reverse boundary primer	gcttcaaggatatgggtagaaaa
bndrp-rpmA	rpmA reverse boundary primer	gcatttttaccggttatcgaatg
bndrp-rpsS	rpsS reverse boundary primer	caacaaggcgaaccttctg
bndrp-rpsN	rpsN reverse boundary primer	cgttacggatacgggtca
bndrp-rplU	rplU reverse boundary primer	tctgaatcgcgaccgtta
bndrp-rpsJ	rpsJ reverse boundary primer	acgggtcatacccacttt
bndrp-rplX	rplX reverse boundary primer	aactcagtcatgagtttttaac
bndrp-rplW	rplW reverse boundary primer	ttaaccactttaactacgtggc
bndrp-rplV	rplV reverse boundary primer	caggtagagttccatggttttac
bndrp-rplS	rplS reverse boundary primer	caccagcaaacagataaaaaagg
bndrp-rplR	rplR reverse boundary primer	gtttaccgcgatcagcttt
bndrp-rplT	rplT reverse boundary primer	ctacggcgataaaagtcaatgt
bndrp-rpsM	rpsM reverse boundary primer	ccgtcagagacttgttttctt
bndrp-rplL	rplL reverse boundary primer	tacagcgcaaaaaggctg
bndrp-rplN	rplN reverse boundary primer	tttaccgcgtttacctttatctt
bndrp-rpsL	rpsL reverse boundary primer	ttcaggattgtccaaaactctac
bndrp-rplQ	rplQ reverse boundary primer	cagctattgtagataagtgggga
bndrp-rpsK	rpsK reverse boundary primer	gctcagcttgagcttagga
bndrp-rpsI	rpsI reverse boundary primer	tttacgctgattcagattttagc
bndrp-rpsH	rpsH reverse boundary primer	gttgatttttacgtcaacgcc
bndrp-rplM	rplM reverse boundary primer	cgagctgcggaacttttg

Table S6-8 (Continued).		
bndrp-rplP	rplP reverse boundary primer	caggttgaactgctcacg
bndrp-rplO	rplO reverse boundary primer	cagcagtctgcgtttcag
bndrp-rplJ	rplJ reverse boundary primer	gtgatagacatttaaattgttcc
bndrp-rpsE	rpsE reverse boundary primer	agcgttgccttgtgtttc
bndrp-rplF	rplF reverse boundary primer	ccagctcctggagcttgc
bndrp-rplE	rplE reverse boundary primer	gtttcgcgaagtatttatcagc
bndrp-rpsG	rpsG reverse boundary primer	cactgataccgatgttacgg
bndrp-rpsD	rpsD reverse boundary primer	tcgaactcacttgctcgata
bndrp-rplD	rplD reverse boundary primer	tggatttttccatcgcagtag
bndrp-rplC	rplC reverse boundary primer	ttcgttgaaatcacgaccg
bndrp-rpsC	rpsC reverse boundary primer	taacatccgtaccctgcg
bndrp-rpsB	rpsB reverse boundary primer	cggtcacttactgatgtaagc
bndrp-rplB	rplB reverse boundary primer	ctttctctaccttcttcagcaa
bndrp-prfB	prfB reverse boundary primer	cgacgcgttttcagttca
bndrp-rpsA	rpsA reverse boundary primer	tgcttgattacaggacgaaac
natfp-rpmH	rpmH forward natural sequence primer	ctgtactgaagcgcaacc
natfp-rpmD	rpmD forward natural sequence primer	cagtgcaatcggtcgtct
natfp-rpmC	rpmC forward natural sequence primer	gagcgttgaagagctgaac
natfp-rpsR	rpsR forward natural sequence primer	aagttctgccgtttcacc
natfp-rpmB	rpmB forward natural sequence primer	ccaagttactggcaagcg
natfp-rpsP	rpsP forward natural sequence primer	cgctaaaaagcgtccgttc
natfp-rpsQ	rpsQ forward natural sequence primer	gcaaggtcgcgttgttag
natfp-rpmA	rpmA forward natural sequence primer	gtaacggtcgcgattcag
natfp-rpsS	rpsS forward natural sequence primer	gtccttttattgacctgcact
natfp-rpsN	rpsN forward natural sequence primer	tgaaagcacgcgaagtaaaa
natfp-rplU	rplU forward natural sequence primer	acaacaccgagtaagcga
natfp-rpsJ	rpsJ forward natural sequence primer	tgaaagcgtttgatcatcgt
natfp-rplX	rplX forward natural sequence primer	gtgatgacgaagttatcgtgtta
natfp-rplW	rplW forward natural sequence primer	aacgtctgctgaaggtgc
natfp-rplV	rplV forward natural sequence primer	catgctcgttcttctgctc
natfp-rplS	rplS forward natural sequence primer	acttgaacaagagcagatgaag
natfp-rplR	rplR forward natural sequence primer	ctgctcgtatccgtcgtg
natfp-rplT	rplT forward natural sequence primer	cgtgcacgtcacaagaaa
natfp-rpsM	rpsM forward natural sequence primer	ctgatcataagcatgccgtaa
natfp-rplL	rplL forward natural sequence primer	ttgaagcagttgcagctatg
natfp-rplN	rplN forward natural sequence primer	gactatgctgaacgtcgc
natfp-rpsL	rpsL forward natural sequence primer	accagctggtacgcaaac

Table S6-8 (Continued).		
natfp-rplQ	rplQ forward natural sequence primer	tcaactgaaccgcaacag
natfp-rpsK	rpsK forward natural sequence primer	cacgtaaacgtgtaagaaaacaa
natfp-rpsI	rpsI forward natural sequence primer	atactacggcactggtcg
natfp-rpsH	rpsH forward natural sequence primer	gctgacccgtatccgtaa
natfp-rplM	rplM forward natural sequence primer	agaaaccgtaaaacgcgac
natfp-rplP	rplP forward natural sequence primer	aaattccgtaaaatgcacaaagg
natfp-rplO	rplO forward natural sequence primer	ccgaaggctccaaaaagg
natfp-rplJ	rplJ forward natural sequence primer	acaagcgattgttgctgaa
natfp-rpsE	rpsE forward natural sequence primer	ggcgaactgcaggaaaag
natfp-rplF	rplF forward natural sequence primer	taaagcaccggtcgttgt
natfp-rplE	rplE forward natural sequence primer	ctgcatgattactacaaagacga
natfp-rpsG	rpsG forward natural sequence primer	tcagcgtaaaattctgccg
natfp-rpsD	rpsD forward natural sequence primer	taagctcaagctgagccg
natfp-rplD	rplD forward natural sequence primer	gagcgcgctgactgtttc
natfp-rplC	rplC forward natural sequence primer	gtgggtatgacccgtatctt
natfp-rpsC	rpsC forward natural sequence primer	atggtattcgcctgggtatt
natfp-rpsB	rpsB forward natural sequence primer	caaggctggtgttcacttc
natfp-rplB	rplB forward natural sequence primer	gttaaatgtaaaccgacatctcc
natfp-prfB	prfB forward natural sequence primer	ttcaggacctcacggaac
natfp-rpsA	rpsA forward natural sequence primer	ctcaactctttgaagagtcctt
natrp-rpmH	rpmH reverse natural sequence primer	ggcctttagcacgacgac
natrp-rpmD	rpmD reverse natural sequence primer	ggaaaccgcgttgatcatac
natrp-rpmC	rpmC reverse natural sequence primer	agtcttaacgcgtgcgac
natrp-rpsR	rpsR reverse natural sequence primer	tcagtgtacggcagcagg
natrp-rpmB	rpmB reverse natural sequence primer	gttcagccagaactgtatcg
natrp-rpsP	rpsP reverse natural sequence primer	cagcaacgcgatcagaaa
natrp-rpsQ	rpsQ reverse natural sequence primer	aaccagcgtccaggattt
natrp-rpmA	rpmA reverse natural sequence primer	ggtttttcgggcctttaact
natrp-rpsS	rpsS reverse natural sequence primer	ggccgcgataagtacgag
natrp-rpsN	rpsN reverse natural sequence primer	ggatttcaccgcgcatag
natrp-rplU	rplU reverse natural sequence primer	atgccagtaattttcacatcagt
natrp-rpsJ	rpsJ reverse natural sequence primer	atctgcacgtctacaccg
natrp-rplX	rplX reverse natural sequence primer	gttagacttgaagaaacggactt
natrp-rplW	rplW reverse natural sequence primer	tctggccttctttcaggg
natrp-rplV	rplV reverse natural sequence primer	acagtgatgtggctggtg
natrp-rplS	rplS reverse natural sequence primer	cagccttaccagtacgct
natrp-rplR	rplR reverse natural sequence primer	atctgccagtgcctggac

Table S6-8 (Continued).		
natrp-rplT	rplT reverse natural sequence primer	ttcaaccagagcggtgaa
natrp-rpsM	rpsM reverse natural sequence primer	ttacgggtacgtgcgttg
natrp-rplL	rplL reverse natural sequence primer	cagcttcttccagagcttttt
natrp-rplN	rplN reverse natural sequence primer	ggtgccagagagataattttcat
natrp-rpsL	rpsL reverse natural sequence primer	catacttggaacgagcctg
natrp-rplQ	rplQ reverse natural sequence primer	tgctttctctgaacgatcaac
natrp-rpsK	rpsK reverse natural sequence primer	aaccgttatgagggatcgg
natrp-rpsI	rpsI reverse natural sequence primer	gacgtgctttacgcagac
natrp-rpsH	rpsH reverse natural sequence primer	cagataatttcgccaccaaga
natrp-rplM	rplM reverse natural sequence primer	gcgtggttgtgctcgtta
natrp-rplP	rplP reverse natural sequence primer	ttaatcggcagtttcgctg
natrp-rplO	rplO reverse natural sequence primer	gcttcgatagcagcacga
natrp-rplJ	rplJ reverse natural sequence primer	cagccagagtacgaacca
natrp-rpsE	rpsE reverse natural sequence primer	ttcttcaacggatttaccacg
natrp-rplF	rplF reverse natural sequence primer	cttcgtcggcgtaacgaa
natrp-rplE	rplE reverse natural sequence primer	gaagtcaaaggcagccag
natrp-rpsG	rpsG reverse natural sequence primer	aaagcgggctgcttactg
natrp-rpsD	rpsD reverse natural sequence primer	cgatcaggtgttcgttaatgtc
natrp-rplD	rplD reverse natural sequence primer	cttaacagcatcagcagtcatt
natrp-rplC	rplC reverse natural sequence primer	cagctggtttaacgatcagg
natrp-rpsC	rpsC reverse natural sequence primer	ttacgctgctgctttttagg
natrp-rpsB	rpsB reverse natural sequence primer	gaagctttcttccgcctg
natrp-rplB	rplB reverse natural sequence primer	ttatcagtacgcttgttgctg
natrp-prfB	prfB reverse natural sequence primer	gctttcaaacttgcttcgataaa
natrp-rpsA	rpsA reverse natural sequence primer	cttcagccattgcgttgt
synfp-rpmH	rpmH forward recoded sequence primer	gtgttttgaaacgtaatcgctc
synfp-rpmD	rpmD forward recoded sequence primer	acacagactcgttctgctatt
synfp-rpmC	rpmC forward recoded sequence primer	cgaaaaatctgtggaggaactaa
synfp-rpsR	rpsR forward recoded sequence primer	aattttgtcgctttacggct
synfp-rpmB	rpmB forward recoded sequence primer	gggaaacgcccagttaca
synfp-rpsP	rpsP forward recoded sequence primer	gcgaagaaacgcccatttt
synfp-rpsQ	rpsQ forward recoded sequence primer	acgtgtggtgtcggataa
synfp-rpmA	rpmA forward recoded sequence primer	gtggatcaactcgcaatgg
synfp-rpsS	rpsS forward recoded sequence primer	cgttcatcgatttgcatctgt
synfp-rpsN	rpsN forward recoded sequence primer	gtgaggtgaagcgagttg
synfp-rplU	rplU forward recoded sequence primer	cagcatcgtgtttcagagg
synfp-rpsJ	rpsJ forward recoded sequence primer	gactaaaggctttcgaccac

Table S6-8 (Continued).		
synfp-rplX	rplX forward recoded sequence primer	gatgaggtaattgttctgacgg
synfp-rplW	rplW forward recoded sequence primer	gcgcttgttgaaagtattgc
synfp-rplV	rplV forward recoded sequence primer	gcgatcaagtgcacaaaaag
synfp-rplS	rplS forward recoded sequence primer	agcaggaacaaatgaaacaaga
synfp-rplR	rplR forward recoded sequence primer	tacacgtgctcgtcgtaa
synfp-rplT	rplT forward recoded sequence primer	gcgctcgccataaaaaga
synfp-rpsM	rpsM forward recoded sequence primer	atatcccggaccacaaaca
synfp-rplL	rplL forward recoded sequence primer	ccagattatcgaggcggt
synfp-rplN	rplN forward recoded sequence primer	aaatgtagctgataatagtgggg
synfp-rpsL	rpsL forward recoded sequence primer	aatcaattagttcgtaagcctcg
synfp-rplQ	rplQ forward recoded sequence primer	cgccagttaaatcgtaattcatc
synfp-rpsK	rpsK forward recoded sequence primer	aagcgcgttcgtaagcag
synfp-rpsI	rpsI forward recoded sequence primer	cgtcgtaagtcaagtgctg
synfp-rpsH	rpsH forward recoded sequence primer	ctgacatgttaacgcgca
synfp-rplM	rplM forward recoded sequence primer	aacggtgaagcgtgattg
synfp-rplP	rplP forward recoded sequence primer	tttcgcaagatgcataaggg
synfp-rplO	rplO forward recoded sequence primer	tgcagagggaagcaagaa
synfp-rplJ	rplJ forward recoded sequence primer	gcaggataagcaggcaatc
synfp-rpsE	rpsE forward recoded sequence primer	gggtgagttacaagagaaattga
synfp-rplF	rplF forward recoded sequence primer	aaggctcctgtagtggtg
synfp-rplE	rplE forward recoded sequence primer	ataaggatgaggtggtgaagaa
synfp-rpsG	rpsG forward recoded sequence primer	gcaagatcttaccagaccct
synfp-rpsD	rpsD forward recoded sequence primer	attatctcgccgcgaagg
synfp-rplD	rplD forward recoded sequence primer	agtgcactaacggtatctgaa
synfp-rplC	rplC forward recoded sequence primer	atgacgcgcatttttactga
synfp-rpsC	rpsC forward recoded sequence primer	atccgtttgggcatcgtg
synfp-rpsB	rpsB forward recoded sequence primer	aagcaggcgtacattttgg
synfp-rplB	rplB forward recoded sequence primer	gcaagcctacgtcacctg
synfp-prfB	prfB forward recoded sequence primer	acaaccgtatccaagatttaaca
synfp-rpsA	rpsA forward recoded sequence primer	aggaaagcctgaaggagatt
synrp-rpmH	rpmH reverse recoded sequence primer	gttaagcgcgcacgacc
synrp-rpmD	rpmD reverse recoded sequence primer	tactgcattaatcattccacgg
synrp-rpmC	rpmC reverse recoded sequence primer	tgtttttacacgcgccac
synrp-rpsR	rpsR reverse recoded sequence primer	acggtccgtataaggtagtaaa
synrp-rpmB	rpmB reverse recoded sequence primer	gctagcaccgtgtcaatc
synrp-rpsP	rpsP reverse recoded sequence primer	cttaatcaatgctgccacac
synrp-rpsQ	rpsQ reverse recoded sequence primer	ccaatgtccaagacttcgttt

Table S6-8 (Continued).		
synrp-rpmA	rpmA reverse recoded sequence primer	attctttggtcccttcacct
synrp-rpsS	rpsS reverse recoded sequence primer	atgcccacggtacgttcg
synrp-rpsN	rpsN reverse recoded sequence primer	ggaatctcgcctcgcatc
synrp-rplU	rplU reverse recoded sequence primer	gatctttacgtccgtaaaccatt
synrp-rpsJ	rpsJ reverse recoded sequence primer	taatttgaacatccacgccc
synrp-rplX	rplX reverse recoded sequence primer	aaaaaagcgcaccttcttg
synrp-rplW	rplW reverse recoded sequence primer	atccaagttttgaccctcct
synrp-rplV	rplV reverse recoded sequence primer	acaaccgtaatatgagatgtacg
synrp-rplS	rplS reverse recoded sequence primer	cgcttttcctgtgcgttc
synrp-rplR	rplR reverse recoded sequence primer	cgtcagccaacgcttgta
synrp-rplT	rplT reverse recoded sequence primer	ctactaacgccgtaaatgcc
synrp-rpsM	rpsM reverse recoded sequence primer	gagtgcgagcattcgttt
synrp-rplL	rplL reverse recoded sequence primer	ctccaacgccttcttcaac
synrp-rplN	rplN reverse recoded sequence primer	ccaagctaatgatcttcataaat
synrp-rpsL	rpsL reverse recoded sequence primer	gtatttgctgcgcgcttg
synrp-rplQ	rplQ reverse recoded sequence primer	cgatcggtccaccaattca
synrp-rpsK	rpsK reverse recoded sequence primer	ccattgtgcggaattggc
synrp-rpsI	rpsI reverse recoded sequence primer	gcttactaaattgaggtcggc
synrp-rpsH	rpsH reverse recoded sequence primer	aaatgatctcaccgccca
synrp-rplM	rplM reverse recoded sequence primer	ctgttgcgctgcatgatta
synrp-rplP	rplP reverse recoded sequence primer	gtcgtcttgattggcaactta
synrp-rplO	rplO reverse recoded sequence primer	acccgccgcctcaatt
synrp-rplJ	rplJ reverse recoded sequence primer	gctaatgtgcgcactaact
synrp-rpsE	rpsE reverse recoded sequence primer	atctcctccacgctctttc
synrp-rplF	rplF reverse recoded sequence primer	acctcatcagcatatcgca
synrp-rplE	rplE reverse recoded sequence primer	caataatgcacgtccctcc
synrp-rpsG	rpsG reverse recoded sequence primer	tgttttgatgacgcaccag
synrp-rpsD	rpsD reverse recoded sequence primer	gctcattgatatctgcgctt
synrp-rplD	rplD reverse recoded sequence primer	tctacctgtttcacagcgt
synrp-rplC	rplC reverse recoded sequence primer	aatcactacccgtcgctc
synrp-rpsC	rpsC reverse recoded sequence primer	gtttctttggttgcgctg
synrp-rpsB	rpsB reverse recoded sequence primer	ttggcttgctaagtcttgtg
synrp-rplB	rplB reverse recoded sequence primer	aaacttgtctgtgcgtttatttg
synrp-prfB	prfB reverse recoded sequence primer	tcaatgaactggtccaaactc
synrp-rpsA	rpsA reverse recoded sequence primer	cttaaatgcctccgccatc

Table S6-8 (Continued).		
kanR.seqOUT-Nr2	Primer for sequencing the C-terminus of recoded essential genes (hybridizes near the N-terminus of kanR and faces toward the recoded essential gene)	gaatttaatcgcggcctc
3502900.tolC-f	forward primer for generating tolC insertion cassette at nt 3502900	gcgcgttgaattttacatcccgtacgttcccctcaccctaaccctctcccttgaggcacattaacgcc
3502901.tolC-r	reverse primer for generating tolC insertion cassette at nt 3502900	atagcaccgtcaagctaaattccgtactgaacggtcccctcgcccctttgtctagggcggcggatt
4427600.tolC-f	forward primer for generating tolC insertion cassette at nt 4427600	tgaagtcgaactgctggaaatcctctaagcagcgcattctgttcccctcgttgaggcacattaacgcc
4427601.tolC-r	reverse primer for generating tolC insertion cassette at nt 4427600	cgtttggcaaactgaagggtttattgctgaatgcctgctccccctctcgtttctagggcggcggatt
3502822.seq-f	forward primer for screening tolC insertion at nt 3502900	cattaaccgtaggccggataaga
3503081.seq-r	reverse primer for screening tolC insertion at nt 3502900	tcccgccgctcttttatcg
4427507.seq-f	forward primer for screening tolC insertion at nt 4427600	ctggcatatggcgagc
4427776.seq-r	reverse primer for screening tolC insertion at nt 4427600	tcgatattaggtaacaatacgcgg
tolC.90.del	deletes endogenous tolC	gaatttcagcgacgtttgactgccgtttgagcagtcatgtgttaaagcttcggccccgtctgaacgtaaggcaacgtaaagatacgggttat
tolC-r_null_mut*	inactivates tolC for CoS-MAGE	a*g*caagcacgccttagtaacccggaattgcgtaagtctgccgctaaatcgtgatgctgcctttgaaaaaattaatgaagcgcgcagtcca
tolC-r_null_revert*	tolC co-selection oligo	c*a*gcaagcacgccttagtaacccggaattgcgtaagtctgccgccgatcgtgatgctgcctttgaaaaaattaatgaagcgcgcagtcca
bla_mut*	inactivates bla in lambda prophage	g*c*c*a*catagcagaactttaaaagtgctcatcattggaaaacgttattaggggcgaaaactctcaaggatcttaccgctgttgagatccag
bla_restore*	bla co-selection oligo	g*c*c*a*catagcagaactttaaaagtgctcatcattggaaaacgttcttcggggcgaaaactctcaaggatcttaccgctgttgagatccag
rplJ_12-54	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*c*tactgcagacagcgcgcctttcgctacttcgcttacttcagcaacaatcgcttgtttgtcttgcagatttaaagccattagctttgct
rplJ_42-87	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*g*tcattttatctacagttacgccacgcgaatccgcaactactgcagacagcgcgcctttcgctacttcgcttacttcagcaacaatcgc
rplJ_321-333	CoS-MAGE oligo that simultaneously changes multiple nearby codons	g*g*tcgatctgagacgccgggatcagctcaccttcaaacgcagccgcttttacctcaaattttgcattcgctttcgcgaactctttgaaca
rplJ_390-423	CoS-MAGE oligo that simultaneously changes multiple nearby codons	c*c*agtttgccagccgaagcttctttcatcgttgccatcaggcgtgcaattgcttcttcgtacgtcggcagagttgccaggcggtcgatct
rpsB_12-57	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*t*tttcgggttccagtaacgcgtctggtgaccgaagtgaacaccagccttcagcatgtcgcgcatagaaacagttgccatgattaaaacc
rplB_147-117	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*g*cttgtggccaccaccgatatgacgagtcgtgatacggccattgttgttacgaccaccacttttgctgtttttttccagcaacggagca
rplB_240-261	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*c*cgtctttgtacagaaccagcgcgatgttcgccgaacggttcggatcgtactccagacgttcaacaactgccgggataccgtctttgtt
rplB_468-516	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*c*accagaacgcagacgcagcgtcacataagcaccatcacgagcaacgatctgaacgtaagtaccagcactacgtgccagctgaccgcct
rplB_654-666	CoS-MAGE oligo that simultaneously changes multiple nearby codons	c*g*accttcaccaccaccatgtgggtggtctaccgggttcatcgccgtaccgcgaacagtcggacgaacaccacgccagcgtgcagcacct
rplB_753-768	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*c*gtacgatgaatttatcagtacgcttgttgctgcgagtcttcttaccttcgtctgaacgccccacggagttaccgggtgcttaccaaa

Table S6-8 (Continued).		
rplD_42-51	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*a*cctggtgaaccagcgcttcgttgaaatcacgaccgaatgtagtttcagaaacagtcagcgcgctctgcgcgtctttcaatactaattc
rplD_162-192	CoS-MAGE oligo that simultaneously changes multiple nearby codons	g*g*gctcttgatagaaccagaacgcgcacggccagtgccttttctggcgccacggtttttttacctgaaccagttacttcagcacgagtcttc
rplD_327-360	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*t*tcggcgcttctacagagaacttctctacaacgatcagacgatcctgacgtaccagttccgacaggatgcttttcagcgcgccgcggta
rplD_543-603	CoS-MAGE oligo that simultaneously changes multiple nearby codons	c*g*ttcttcacgaattatgccagcatctcctcaacttgcttaacagcatcagcagtcattacaactttgtcgaacgcgatcaggctaaccg
rplO_24-33	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*g*aaccgataccacgacccaggcgtttacccgccttttgctgccttctgccggagacagagtatttaaacgcatctcttactcctcaac
rplO_78-87	CoS-MAGE oligo that simultaneously changes multiple nearby codons	g*c*cagaacgagacttctgacctttgtgaccacgaccacccgttttaccccaggccagaaccgataccacgacccaggcgtttacccgcctt
rplQ_159-201	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*c*gatctcgttatcacgagtacgagcgaatgccagacgacgattagcaacgctatcagtcttagccagagtaatcagcggctcaactacg
rpsC_60-75	CoS-MAGE oligo that simultaneously changes multiple nearby codons	c*t*ttaaaatcgctgtccaggttgtcagcgaattcttttgtgttcgcaaaccaagtagagttccatggttttacaatacccaggcgaatac
rpsC_150-198	CoS-MAGE oligo that simultaneously changes multiple nearby codons	c*c*gggcgagcagtgtgaatagttacacggatgctcttagccggacgctcgataacgatacgagatactgacgctttagccagttccttag
rpsC_267-309	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*c*cagttcaggcttacgaacttcagcgatgttgatctgtgcaggaacgccagcgatgtccgctactaccttacgcagtttttctacgtct
rpsD_6-60	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*c*ttcaggaataagtccgtgccctcacgacggctcagcttcagcttaggacccaaatagcgtgccattttctttctccaacaaacctgga
rpsD_48-87	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*g*ctggccaggagcttgttcaattttacacttagtatcgatcgcgcgaacgccagacttcaggaataagtccgtgccctcacgacggctc
rpsG_144-195	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*g*tcgggcgcacgttttccagagctacttcgaatgcttccagttcagatttaccagagcgctgagccagtgtctccagcgcgctgtatac
rpsG_342-369	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*a*ctgcagtacctttgttttctgcagcatcagacagttcgttcgccaggcgcagagccatgcttttatcaccgcgtttacgagcagcttc
rpsG_432-438	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*g*cgcagagataaccaacggtagtgtgcgaacgccttgtttgcttcagccatacggtgaacgtcttcacgtttcttaactgcagtacctt
rpsG_471-516	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*t*ttaagtagcccaaagccggctgcttacttgaagcgcccgcctggtgactaaaactgcgcagagataaccaacggtagtgtgcgaacgc
rpsG_504-537	CoS-MAGE oligo that simultaneously changes multiple nearby codons	g*c*gatgggtgttgtacgagccatttgtttcctcgtttatcttttaggcgttaatttaagtagcccaaagccggctgcttacttgaagcg
rpsA_21-51	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*c*aacgccacgaacgatagaacccgggcgcgtttcgatttcttttaagctctcttcaaataattgagcaaaagattcagtcatgtttaat
rpsA_135-165	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*c*acctacctggatttccagctcgccctgagcgtttttgaactgctcagccgggatcgcgctctcagatttcagaccagcgtcaaccagt
rpsA_252-324	CoS-MAGE oligo that simultaneously changes multiple nearby codons	c*c*agtaacagtttcagcatcttcgtaagctttttccagcgtgatccaagcttcgtgacgtttagctttctcacggctcagcagagtttca
rpsA_453	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*c*aacgttgttgcgcttctgatccagcttgattactttaaattccagctctttgccttccaggtgcagagtgtcacgcaccggacgaacg
rpsA_513-525	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*g*gttttccagcagctgatcgcgctctgcgctgttttccgattcgataacagcacgacgagaaacaacaacgttgttgcgcttctgatcc
rpsA_603	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*a*cgccgcccagatcaacgaatgcaccgtagtcagtcaggttcttaacgataccttaacttccatgccttcctg

Table S6-8 (Continued).		
rpsA_669-702	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*t*taacagtgatttcgtcgcccacgttaacgatttcgctcggatgcttaacgcgtttccaagccatgtcagtgatgtgcagcaggccgtc
rpsA_756-765	CoS-MAGE oligo that simultaneously changes multiple nearby codons	g*a*tagctacccacggatcttcgcccagctgtttcaggcccagtgatacacgagtacgttcgcggtcgaacttcagcactttaacagtgat
rpsA_831-861	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*c*ttcgatttcaacgaagcagccgtagtcagtcaggttagtcacgcgaccagtcagtttcgtaccttccggataacgtttagcgatagct
rpsA_918-954	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*c*tacatcgccaacgttaacaactttactcgggtggatgtttttgttagtccagtccatttcgctaacgtgtaccaggccttcaacgcct
rpsA_1026-1077	CoS-MAGE oligo that simultaneously changes multiple nearby codons	a*c*ggtcgcccttgttgtgcgtttccgcgaactgctgccacgggttagctttgcactgtttcagacccagtgagatacgacgacgttcttc
rpsA_1188	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*t*ttttgtattcacgaactgcttcttcgcctgcaacgttccaagagatgtcagacaggtgaaccaggccgtcgatgccgccgtccaagcc
rpsA_1287-1305	CoS-MAGE oligo that simultaneously changes multiple nearby codons	c*t*tgttcagagcaacccagttgttgaacggatcttctgctaactgttaacgcccagagagatacgttcacgttctgcgtcaacctgcag
rpsA_1362-1398	CoS-MAGE oligo that simultaneously changes multiple nearby codons	c*t*tcaacgccgtcagccagttctaccgttgcgcctttagcgtcaactgcagttactttaccagttacgatagcgcctttcttgttcagag
rpsA_1449-1515	CoS-MAGE oligo that simultaneously changes multiple nearby codons	g*a*tcaacgcccgtgaatttagcttcaacttcgtcgccaacgctcagaaccagcgtagcgtcttcaacgcggtcacgcgatgcttcagaag
rpsA_1626	CoS-MAGE oligo that simultaneously changes multiple nearby codons	t*t*actcgcctttagctgctttgaaagcttcagccattgcgttgttactgaagtttgcatcttcctgtttgttaacagttgcgattgcatc
rplJ_ACC171ACG	CoS-MAGE oligo that changes one codon	g*a*ccaacaaacgcgtctttcaggcactcgaacggagtaccttcaacagcacggcgcagcagcgtgttacgaacaacacgcatgtatacgc
rplJ_ACC237ACG	CoS-MAGE oligo that changes one codon	a*g*acgagcagcagcgcccgggtgttccatagagtatgcaatcagcgtcggaccaacaaacgcgtctttcaggcactcgaacggagtacct
rpsR_ACC39ACG	CoS-MAGE oligo that changes one codon	a*g*cgtagcgatatctttatagtcgatctcttgaacgccttccgccgtgaaacggcagaacttgcgacgacggaaataacgtgccatatgg
rplL_TCC99TCA	CoS-MAGE oligo that changes one codon	g*c*agcttcaaccgggccagcagctacagctacagcagcagcagctgaaacaccgaatttttcttccattgcagagatcagttctacaacg
rpsP_GTC60GTT	CoS-MAGE oligo that changes one codon	t*t*gaagaaaccaacgcgctcgatgaagcgaccgttgcgtgcattacggctgtcagcaacaacaacctggtagaacggacgctttttagcg
rplB_ACC384ACA	CoS-MAGE oligo that changes one codon	a*c*gttatgaacagtagaaccaaccgggatgttgcgcatcggcagtgtgttacctggtttgattgcagcatcaacgccagactgaatctgg
rplB_GCC315GCG	CoS-MAGE oligo that changes one codon	c*a*tcaacgccagactgaatctggtcgccagctttcaggcctttaggcgccaggatgtaacggcgttcaccgtctttgtacagaaccagcg
rplD_ACC249ACG	CoS-MAGE oligo that changes one codon	t*t*cttgttaacttttgactgtggtcctgcggacgagcagcaaacgtcacgccaccagaacgccagatcgggctcttgatagaaccagaa
rplD_ACC447ACG	CoS-MAGE oligo that changes one codon	t*g*caggttgcgcgcagccaggaacaggttttcgtccagctcacccgtgatgatcagcacatcttccagagccatgtctttcagtttctgt
rplO_GCC222GCG	CoS-MAGE oligo that changes one codon	a*c*tacaccgccttctactttagccaggtcagacagacgaatttccgctgtaatcgctgctttacgagaagtgaagccgaatttcggcaga
rplQ_GCC270GCG	CoS-MAGE oligo that changes one codon	t*c*gcctgcacggaagccacacttcagaatacgagtgtaaccacccgcacggctcgcgaaacgcgggcccagttcgttaaacagttttgcc
rpsG_GTC15GTT	CoS-MAGE oligo that changes one codon	t*c*tgatccgaacttcggatccggcagaattttacgctgaccaataacgcgacgacgtggcatggaaatactccgttgttaattcaggatt
rpsG_GTC270GTG	CoS-MAGE oligo that changes one codon	t*c*aacgatccaacgcattgccagagcattacgacgaaccggacgcacttcaactggtacctgataagtagaaccaccaacgcggcgagac

Table S6-8 (Continued).		
rplJ_GCC333GCG	redesigned CoS-MAGE oligos to convert remaining forbidden codons	c*a*gagttgccaggcggtcgatctgagacgccgggatcagctcacc ttcaaacgcagcggctttgacctcaaattttgcattcgctttcgc
rplD_TGA603TAA_ref actor	redesigned CoS-MAGE oligos to convert remaining forbidden codons	g*t*gcggtgcacgcagcaccttcagcagacgttcttcacgaatcatta tgccagcatctcctcaacttgcttaacagcatcagcagtcat
rplQ_CTT162CTG	redesigned CoS-MAGE oligos to convert remaining forbidden codons	c*g*ggcgaatgccagacgacgattagcaacgctatcagtcttggcc agagtaatcagcggctcaactacgcggcgcagctctttcgcttta
rpsA_ACC324ACT	redesigned CoS-MAGE oligos to convert remaining forbidden codons	c*a*gctcaacagtgaagccgcccttaactttgccgttgataacacca gtaacagtttcagcatcttcgtaagctttttccagcgtgatcca
rpsL-1	redesigned CoS-MAGE oligos to convert remaining forbidden codons	a*g*tcagacgaacacggcatactttacgcagcgcgctgttcggttttt aggagtcgtagtatatacacgagtacatacgccacgtttttg
rpsL-2	redesigned CoS-MAGE oligos to convert remaining forbidden codons	a*c*cgccacggatcaggatcacactgtgctcctgcaggttgtgacct tcaccaccgatgtaagaagtcacttcgaaaccgttagtcagacg
rpsL-3	redesigned CoS-MAGE oligos to convert remaining forbidden codons	g*c*ttacggtctttaacgccagagcagtctaacgcaccacgtactgtg tggtaacgaacacccggcaggtctttaacacgaccgccacg
rpsL-4	redesigned CoS-MAGE oligos to convert remaining forbidden codons	a*c*gcttcacgccatacttgctacgagcctgcttacggtctttaacgcc agagcagtctaacgcaccacgtactgtgtggtaacgaacacc
rplQ_CTT162YTR*	CoS-MAGE oligo to change rplQ CUU 160-162 to all Leu codons	g*t*acgggcgaatgccagacgacgattagcaacgctatcagtcttg gcyaragtaatcagcggctcaactacgcggcgcagctctttcgct
rplQ_CTT162ATY*	CoS-MAGE oligo to change rplQ CUU 160-162 to all Ile codons	g*t*acgggcgaatgccagacgacgattagcaacgctatcagtcttg gcratagtaatcagcggctcaactacgcggcgcagctctttcgct
rplQ_CTT162GTD*	CoS-MAGE oligo to change rplQ CUU 160-162 to all Val codons	g*t*acgggcgaatgccagacgacgattagcaacgctatcagtcttg gchacagtaatcagcggctcaactacgcggcgcagctctttcgct
rplQ_CTT162GCD*	CoS-MAGE oligo to change rplQ CUU 160-162 to all Ala codons	g*t*acgggcgaatgccagacgacgattagcaacgctatcagtcttg gchgcagtaatcagcggctcaactacgcggcgcagctctttcgct
rplQ_CTT162ATG*	CoS-MAGE oligo to change rplQ CUU 160-162 to the Met ATG codon	g*t*acgggcgaatgccagacgacgattagcaacgctatcagtcttg gccauagtaatcagcggctcaactacgcggcgcagctctttcgct
rplP_syn_fix_G	MAGE oligo to convert rplP_syn1 AUA to AUG, AUC, or AUU	c*g*acacgtcagttccttgagctaggccacgattgcgtcccttatgC atcttgcgaaacttcgttcgtttcggctgcagcatcagcgacgc
rplO_24-33_wt-f	wt forward mascPCR primer	cccgccttttggagccttcg
rplO_78-87_wt-f	wt forward mascPCR primer	ggttctggcctcggtaaaacc
rplO_GCC222GCG_wt -f	wt forward mascPCR primer	cgtaaagcagcgattacagcc
rplQ_159-201_wt-f	wt forward mascPCR primer	gcaacgctatcagtcttggcaag
rplQ_GCC270GCG_wt -f	wt forward mascPCR primer	cgtttcgcgagccgtgcc
rpsC_60-75_wt-f	wt forward mascPCR primer	cgaattctttggtgttcgcaaaccag
rpsC_150-198_wt-f	wt forward mascPCR primer	taaggaactggctaaagcgtcc
rpsC_267-309_wt-f	wt forward mascPCR primer	cctgcacagatcaacatcgcc
rpsD_6-60_wt-f	wt forward mascPCR primer	gcttgagcttaggacccaaatatct
rpsD_48-87_wt-f	wt forward mascPCR primer	gcgttcgcgcgatcgatacc
rpsR_ACC39ACG_wt-f	wt forward mascPCR primer	gtcgcaagttctgccgtttcacc

Table S6-8 (Continued).		
rplL_TCC99TCA_wt-f	wt forward mascPCR primer	gcaatggaagaaaaattcggtgtttcc
rpsP_GTC60GTT_wt-f	wt forward mascPCR primer	gtccgttctaccaggttgttgtc
rpsB_12-57_wt-f	wt forward mascPCR primer	actgtttccatgcgcgacatgctc
rplJ_GCC333GCG-wt-f	wt forward mascPCR primer	ttgaggtcaaagccgctgcc
rplD_TGA603TAA_refactor-wt-f	wt forward mascPCR primer	aagttgaggagatgctggcatg
rplQ_CTT162CTG-wt-f	wt forward mascPCR primer	gtagttgagccgctgattactctt
rpsA_ACC324ACT-wt-f	wt forward mascPCR primer	cttacgaagatgctgaaactgttacc
rplO_24-33_mut-f	mutant forward mascPCR primer	cccgccttttgctgccttct
rplO_78-87_mut-f	mutant forward mascPCR primer	ggttctggcctgggtaaaacg
rplO_GCC222GCG_mut-f	mutant forward mascPCR primer	cgtaaagcagcgattacagcg
rplQ_162-201_mut-f	mutant forward mascPCR primer	gcaacgctatcagtcttagccag
rplQ_GCC270GCG_mut-f	mutant forward mascPCR primer	cgtttcgcgagccgtgcg
rpsC_60-75_mut-f	mutant forward mascPCR primer	cgaattcttttgtgttcgcaaaccaa
rpsC_150-198_mut-f	mutant forward mascPCR primer	taaggaactggctaaagcgtca
rpsC_267-309_mut-f	mutant forward mascPCR primer	cctgcacagatcaacatcgct
rpsD_6-60_mut-f	mutant forward mascPCR primer	cttcagcttaggacccaaatagcg
rpsD_48-87_mut-f	mutant forward mascPCR primer	gcgttcgcgcgatcgatact
rpsR_ACC39ACG_mut-f	mutant forward mascPCR primer	gtcgcaagttctgccgtttcacg
rplL_TCC99TCA_mut-f	mutant forward mascPCR primer	gcaatggaagaaaaattcggtgtttca
rpsP_GTC60GTT_mut-f	mutant forward mascPCR primer	gtccgttctaccaggttgttgtt
rpsB_12-57_mut-f	mutant forward mascPCR primer	actgtttctatgcgcgacatgctg
rplJ_GCC333GCG-mut-f	mutant forward mascPCR primer	ttgaggtcaaagccgctgcg
rplD_TGA603TAA_refactor-mut-f	mutant forward mascPCR primer	aagttgaggagatgctggcata
rplQ_CTT162CTG-mut-f	mutant forward mascPCR primer	gtagttgagccgctgattactctg

Table S6-8 (Continued).		
rpsA_ACC324ACT-mut-f	mutant forward mascPCR primer	cttacgaagatgctgaaactgttact
rplO_24-33_rev	reverse mascPCR primer	tgaacgcgctctggaaaaagg
rplO_78-87_rev	reverse mascPCR primer	gcatctgaccaccctcgaaacc
rplO_GCC222GCG_rev	reverse mascPCR primer	cagcggtgaagtagaatgcaaagc
rplQ_159-201_rev	reverse mascPCR primer	ctgagaaggataaggtcatgcgc
rplQ_GCC270GCG_rev	reverse mascPCR primer	ggtaagcaaccggcattcttcag
rpsC_60-75_rev	reverse mascPCR primer	caagaaagttctggaatctgccattg
rpsC_150-198_rev	reverse mascPCR primer	cgccagcgatgtccgctac
rpsC_267-309_rev	reverse mascPCR primer	aggtgttgtagtcgatgtcagcac
rpsD_6-60_rev	reverse mascPCR primer	ggtccgcgcaaaccgatc
rpsD_48-87_rev	reverse mascPCR primer	aaactctgtcacagaaccctgc
rpsR_ACC39ACG_rev	reverse mascPCR primer	gtttcgagcgagtgccgcag
rplL_TCC99TCA_rev	reverse mascPCR primer	cgaatacgtttttctcggtataggagtaaacc
rpsP_GTC60GTT_rev	reverse mascPCR primer	tgataacctgcccatcgaggaac
rpsB_12-57_rev	reverse mascPCR primer	ccaggctgttttccagtttctcc
rplJ_GCC333GCG-r	reverse mascPCR primer	cctgaatatcagaataagtttatacgtaagcgaatg
rplD_TGA603TAA_refactor-r	reverse mascPCR primer	ccttgtgcagctcagggttaac
rplQ_CTT162CTG-r	reverse mascPCR primer	cgctggagatcgctttcggtatatag
rpsA_ACC324ACT-r	reverse mascPCR primer	caacgaatgcaccgtagtcagt
vsr_mut*	MAGE oligo that inactivates vsr by adding two in-frame stop codons	g*g*c*c*ctgcccggttaacagactggcgaggcgcttctcttatcacgtatcacgcgtggcaatcgcgcgcatatttttgctg
vsr_wt-f	wt forward mascPCR primer	ccacgcgtgatacggc
vsr_mut-f	mutant forward mascPCR primer	gccacgcgtgatacgtg
vsr-r	reverse mascPCR primer	cgcactcccagacaatcaatac
rplP_syn_fix_wt-f	wt forward mascPCR primer	acgaacgaagtttcgcaagatA
rplP_syn_fix_mut-f	mutant forward mascPCR primer	acgaacgaagtttcgcaagatB
rplP_syn_fix_305-r	reverse mascPCR primer	catccatctcgtataataccttgcc
rpsS-Leu-fix-f	Primer converts the rpsS_syn1 forbidden CUU codon to CUA, CUG, UUA, or UUG	GATAAATTCATCGTACGTCGCCGTAGCAAATAATTTTAGAGGATAAGCCATGytrCGhAGCTTAAAAAAGGGACC

Table S6-8 (Continued).		
rpsS-Pro-fix-f	Primer converts the rpsS_syn1 forbidden CUU codon to CCA, CCG, or CCU	GATAAATTCATCGTACGTCGCCGTAGCAAA TAATTTTAGAGGATAAGCCATGccdCGhAGC TTAAAAAAGGGACC
rpsS-Syn-r	Reverse primer for rpsS_syn cassette	AGAACGAGCATGGCGATG

An asterisk (*) indicates a phosphorothioate bond used to protect against exonuclease activity.

Table S6-9. Recoded gene designs

>rpmH
ATGAAGCGTACGTTCCAGCCTAGTGTTTTGAAACGTAATCGCTCACATGGTTTTCGCGCGCGCATGGCAACGAAGAA
CGGCCGCCAAGTGTTGGCGCGCCGCCGCGCAAAGGGTCGTGCGCGCTTAACGGTGTCAAAATAATCTAGAAAGACGT
C

>rpmD
ATGGCGAAAACGATCAAGATCACACAGACTCGTTCTGCTATTGGGCGCTTGCCTAAGCATAAAGCTACATTACTGGG
GTTAGGCTTACGACGCATCGGACATACGGTTGAACGTGAGGACACGCCGGCAATCCGTGGAATGATTAATGCAGTAT
CGTTTATGGTGAAGGTGGAAGAATAATCTAGAAAGACGTC

>rpmC
ATGAAGGCGAAGGAATTGCGCGAAAAATCTGTGGAGGAACTAAATACGGAATTGTTAAATTTATTGCGAGAACAATT
TAATTTACGCATGCAAGCAGCGTCGGGACAATTGCAGCAAAGCCATCTACTGAAACAGGTACGCCGTGACGTGGCGC
GTGTAAAAACACTGTTGAATGAAAAAGCTGGCGCTTAATCTAGAAAGACGTC

>rpsR
ATGGCGCGCTACTTTCGCCGCCGTAAATTTTGTCGCTTTACGGCTGAGGGTGTGCAGGAAATTGATTACAAGGACAT
TGCGACTTTAAAGAATTATATTACGGAGTCTGGCAAAATCGTTCCTTCACGCATTACAGGCACTCGAGCTAAGTATC
AACGCCAATTAGCGCGTGCAATTAAGCGTGCGCGTTATTTAAGTTTACTACCTTATACGGACCGTCACCAATAATCT
AGAAAGACGTC

>rpmB
ATGTCTCGTGTATGTCAGGTGACGGGGAAACGCCCAGTTACAGGCAATAATCGCAGCCATGCTCTAAATGCTACGAA
GCGTCGCTTTTTGCCAAATTTACATAGCCATCGCTTTTGGGTGGAATCAGAAAAACGATTCGTGACTTTACGTGTGA
GCGCAAAGGGCATGCGCGTTATTGACAAGAAGGGGATTGACACGGTGCTAGCAGAGCTACGCGCGCGCGGTGAGAAA
TATTAATCTAGAAAGACGTC

>rpsP
ATGGTGACGATCCGACTGGCTCGCCATGGAGCGAAGAAACGCCCATTTTATCAAGTGGTGGTTGCAGATTCACGAAA
CGCTCGTAATGGCCGTTTTATTGAACGTGTGGGCTTTTTTAATCCGATTGCGTCTGAGAAGGAGGAGGGAACGCGTT
TAGACTTGGACCGTATTGCGCATTGGGTGGGTCAAGGTGCTACGATCTCGGACCGTGTGGCAGCATTGATTAAGGAG
GTGAATAAGGCTGCATAATCTAGAAAGACGTC

>rpsQ
ATGACGGACAAGATTCGCACATTACAGGGACGTGTGGTGTCGGATAAGATGGAAAAGAGCATCGTGGTGGCAATTGA
GCGCTTCGTAAAGCATCCTATTTATGGGAAGTTTATTAAACGCACTACGAAGTTACATGTTCACGATGAAAATAATG
AGTGTGGAATTGGCGATGTAGTGGAGATTCGAGAGTGTCGCCCATTAAGCAAAACGAAGTCTTGGACATTGGTGCGT
GTGGTGGAAAAGGCTGTGTTGTAATCTAGAAAGACGTC

>rpmA
ATGGCGCACAAGAAAGCGGGTGGATCAACTCGCAATGGCCGTGACTCTGAGGCGAAGCGATTGGGGGTAAAACGCTT
TGGCGGTGAGAGTGTGTTAGCTGGGTCAATTATTGTGCGACAGCGCGGAACGAAGTTTCATGCGGGGGCAAATGTGG
GATGTGGCCGCGATCATACGTTATTCGCGAAGGCTGATGGGAAGGTTAAGTTTGAGGTGAAGGGACCAAAGAATCGC
AAGTTCATTTCGATTGAGGCGGAGTAATCTAGAAAGACGTC

>rpsS
ATGCCTCGCAGCTTAAAAAAGGGACCGTTCATCGATTTGCATCTGTTAAAAAAAGTGGAAAAGGCTGTTGAGTCAGG
CGATAAAAAACCGTTGCGTACGTGGTCGCGACGCAGCACAATTTTCCCAAATATGATTGGCCTGACAATTGCGGTGC
ACAACGGACGCCAACATGTACCAGTGTTCGTTACGGATGAGATGGTGGGCCATAAGTTGGGCGAGTTTGCTCCAACG
CGAACGTACCGTGGGCATGCGGCAGACAAGAAGGCAAAAAAAAAAGTAATCTAGAAAGACGTC

>rplW
ATGATTCGCGAGGAGCGCTTGTTGAAAGTATTGCGCGCGCCACATGTGTCAGAGAAGGCTAGTACAGCAATGGAAAA
GAGCAATACGATTGTTTTAAAGGTAGCGAAGGATGCAACAAAGGCGGAGATTAAGGCGGCAGTTCAAAAGTTGTTCG

AGGTAGAGGTGGAGGTAGTGAATACGTTAGTGGTGAAGGGCAAGGTGAAGCGACATGGCCAACGCATTGGACGCCGC
TCAGATTGGAAGAAGGCGTATGTGACTTTGAAGGAGGGTCAAAACTTGGATTTTGTAGGAGGTGCAGAATAATCTAG
AAAGACGTC

>rpsN
ATGGCAAAACAGAGCATGAAGGCGCGTGAGGTGAAGCGAGTTGCGCTGGCAGACAAGTATTTTGCTAAGCGTGCAGA
GCTAAAGGCTATTATTAGCGACGTTAATGCGTCGGATGAGGACCGCTGGAATGCTGTGTTAAAATTGCAAACATTGC
CTCGCGACAGCTCTCCATCGCGACAACGCAATCGTTGTCGCCAGACTGGCCGACCACACGGCTTTTTGCGTAAATTT
GGCCTGAGTCGCATCAAAGTTCGCGAGGCGGCGATGCGAGGCGAGATTCCAGGCCTAAAGAAAGCATCGTGGTAATC
TAGAAAGACGTC

>rplU
ATGTATGCAGTGTTTCAGTCTGGCGGGAAGCAGCATCGTGTTTCAGAGGGGCAAACTGTGCGTTTGGAGAAATTGGA
TATTGCTACGGGAGAAACGGTGGAATTTGCAGAGGTTTTAATGATTGCGAATGGCGAGGAGGTGAAGATTGGTGTGC
CGTTTGTGGACGGTGGTGTTATTAAGGCGGAGGTGGTAGCACATGGGCGCGGAGAAAAGGTGAAGATTGTGAAATTC
CGACGACGAAAGCATTACCGAAAACAACAAGGGCACCGCCAATGGTTTACGGACGTAAAGATCACAGGTATTTCGGC
GTAATCTAGAAAGACGTC

>rpsJ
ATGCAAAATCAGCGCATTCGAATTCGACTAAAGGCTTTCGACCACCGATTGATTGACCAGGCTACGGCAGAGATTGT
GGAAACAGCGAAACGTACGGGCGCACAAGTTCGCGGCCCTATTCCTCTACCAACGCGTAAGGAACGTTTTACAGTAC
TAATTAGTCCTCATGTGAATAAGGATGCTCGTGACCAATATGAGATTCGCACGCATCTGCGATTGGTAGATATTGTG
GAACCTACGGAAAAGACGGTAGACGCATTGATGCGCCTAGACCTAGCAGCGGGCGTGGATGTTCAAATTAGTTTGGG
CTAATCTAGAAAGACGTC

>rplX
ATGGCTGCAAAAATTCGCCGCGACGATGAGGTAATTGTTCTGACGGGAAAGGACAAGGGCAAGCGAGGCAAGGTAAA
AAACGTGTTGAGCAGCGGAAAAGTTATCGTGGAGGGCATTAATCTAGTGAAAAAGCACCAAAAACCTGTACCTGCAT
TGAATCAGCCAGGCGGTATTGTGGAGAAGGAGGCGGCGATCCAAGTGAGCAATGTTGCGATTTTTAACGCAGCGACG
GGAAAAGCAGATCGCGTTGGTTTCCGTTTTGAGGATGGCAAGAAGGTGCGCTTTTTTAAAAGCAATTCGGAAACGAT
TAAATAATCTAGAAAGACGTC

>rplV
ATGGAGACAATTGCAAAGCACCGTCACGCGCGATCAAGTGCACAAAAAGTACGTCTGGTGGCGGATTTGATCCGTGG
GAAAAAGGTTAGCCAAGCGTTAGACATCCTGACTTATACGAATAAAAAGGCAGCGGTGTTAGTAAAAAAGGTGTTAG
AGTCGGCGATCGCGAATGCGGAGCATAATGACGGAGCGGATATCGATGACTTAAAGGTGACTAAGATCTTTGTGGAT
GAGGGTCCTAGTATGAAACGTATCATGCCTCGCGCGAAGGGCCGCGCGGACCGTATTTTAAAACGTACATCTCATAT
TACGGTTGTGGTAAGCGACCGATAATCTAGAAAGACGTC

>rplS
ATGTCAAATATCATCAAACAGCTGGAGCAGGAACAAATGAAACAAGATGTGCCAAGCTTTCGCCCTGGAGACACAGT
AGAGGTAAAGGTGTGGGTGGTAGAGGGGAGCAAGAAGCGCTTGCAAGCTTTTGAAGGTGTAGTGATTGCAATCCGCA
ATCGTGGCTTACATAGCGCGTTTACAGTGCGCAAGATCAGTAATGGTGAGGGGGTGGAACGCGTTTTTCAAACGCAT
AGCCCAGTGGTAGATTCGATCAGCGTAAAGCGCCGAGGCGCGGTGCGCAAGGCGAAGCTATATTATTTGCGCGAACG
CACAGGAAAAGCGGCGCGAATTAAGGAACGCCTGAATTAATCTAGAAAGACGTC

>rplR
ATGGACAAAAAGAGTGCACGAATTCGCCGCGCTACACGTGCTCGTCGTAAACTGCAAGAATTAGGTGCTACACGTTT
AGTAGTGCACCGCACGCCACGCCATATCTATGCGCAAGTGATCGCTCCAAATGGCAGCGAGGTATTGGTGGCGGCGA
GTACGGTTGAGAAGGCGATTGCAGAGCAGTTGAAATATACGGGCAATAAGGATGCTGCGGCGGCGGTAGGCAAGGCG
GTGGCGGAGCGAGCATTGGAGAAGGGTATTAAGGACGTGTCTTTCGATCGAAGTGGCTTTCAGTACCACGGCCGCGT
ACAAGCGTTGGCTGACGCAGCACGCGAGGCGGGACTGCAATTTTAATCTAGAAAGACGTC

>rplT
ATGGCACGTGTGAAGCGCGGCGTGATCGCTCGCGCTCGCCATAAAAAGATCCTGAAGCAGGCGAAGGGGTATTATGG
AGCTCGCTCGCGTGTGTATCGTGTAGCTTTTCAAGCTGTGATTAAGGCGGGGCAATACGCATATCGCGATCGCCGCC
AGCGAAAACGCCAATTTCGCCAGTTATGGATCGCACGCATTAATGCTGCGGCGCGCCAAAATGGCATCAGCTATTCA

AAGTTTATTAACGGTTTAAAGAAGGCAAGCGTGGAGATTGATCGCAAAATTTTAGCGGACATTGCTGTGTTTGATAA
GGTGGCATTTACGGCGTTAGTAGAGAAGGCTAAGGCTGCGTTAGCGTAATCTAGAAAGACGTC

>rpsM
ATGGCTCGCATTGCTGGTATCAATATCCCGGACCACAAACACGCGGTGATTGCTTTAACGTCTATCTACGGAGTGGG
GAAAACTCGAAGTAAGGCAATTTTGGCGGCGGCAGGAATTGCGGAGGACGTGAAAATTAGCGAATTGTCGGAGGGCC
AGATTGATACATTGCGCGATGAGGTGGCGAAGTTCGTTGTAGAGGGCGACTTACGTCGCGAGATTAGTATGTCTATT
AAACGTTTGATGGACCTGGGCTGTTACCGTGGCCTGCGCCACCGACGCGGCCTGCCTGTGCGTGGCCAACGCACTAA
AACGAATGCTCGCACTCGAAAAGGCCCACGTAAGCCTATTAAAAAGTAATCTAGAAAGACGTC

>rplL
ATGTCGATTACGAAGGACCAGATTATCGAGGCGGTAGCGGCGATGAGCGTTATGGATGTGGTGGAGCTAATTTCAGC
GATGGAGGAGAAGTTTGGCGTATCAGCAGCGGCGGCTGTGGCGGTGGCGGCGGGACCTGTGGAGGCGGCGGAGGAAA
AGACAGAGTTTGATGTTATCTTAAAGGCAGCGGGGGCGAATAAGGTAGCGGTAATTAAGGCGGTTCGCGGAGCGACA
GGTTTAGGCTTGAAGGAGGCAAAGGATTTGGTGGAGTCGGCTCCTGCGGCGTTAAAGGAGGGTGTTTCGAAGGATGA
TGCTGAGGCGTTGAAGAAGGCGTTGGAGGAGGCAGGTGCAGAGGTAGAGGTGAAGTAATCTAGAAAGACGTC

>rplN
ATGATTCAGGAGCAAACAATGTTAAATGTAGCTGATAATAGTGGGGCGCGACGTGTGATGTGCATTAAAGTGTTGGG
AGGTAGCCATCGCCGTTATGCGGGTGTTGGTGATATTATTAAAATTACGATTAAGGAGGCGATCCCTCGCGGAAAAG
TTAAGAAGGGCGACGTATTAAAAGCTGTTGTTGTTCGAACAAAAAAAGGCGTGCGTCGTCCAGATGGAAGTGTGATC
CGTTTTGACGGCAACGCATGCGTGCTGCTAAATAATAATTCAGAACAACCGATTGGCACTCGCATCTTCGGCCCTGT
TACGCGCGAATTACGCTCTGAAAAATTTATGAAGATCATTAGCTTGGCGCCGGAGGTGTTATAATCTAGAAAGACGT
C

>rpsL
ATGGCTACAGTAAATCAATTAGTTCGTAAGCCTCGCGCGCGTAAGGTAGCAAAGAGTAATGTTCCGGCATTAGAGGC
TTGTCCTCAGAAGCGCGGGGTTTGCACGCGCGTGTACACGACGACACCGAAGAAGCCAAATAGCGCATTACGAAAGG
TTTGTCGCGTACGCTTAACGAATGGCTTTGAGGTTACGTCTTATATTGGCGGAGAGGGACATAATTTGCAAGAACAT
AGTGTAATTTTGATTCGCGGTGGGCGCGTAAAGGATCTGCCAGGAGTACGCTATCATACAGTGCGAGGGGCTTTAGA
TTGTTCTGGTGTGAAGGATCGAAAACAAGCGCGCAGCAAATACGGTGTTAAACGACCGAAAGCGTAATCTAGAAAGA
CGTC

>rplQ
ATGCGTCACCGCAAATCTGGCCGCCAGTTAAATCGTAATTCATCTCACCGTCAAGCAATGTTTCGTAACATGGCGGG
CTCGTTAGTGCGCCACGAGATTATTAAAACTACATTACCGAAGGCTAAGGAACTACGTCGTGTGGTAGAACCTTTAA
TCACGCTGGCTAAAACGGACTCGGTGGCGAACCGCCGCTTGGCGTTTGCTCGCACGCGCGACAATGAAATTGTTGCT
AAGTTATTCAATGAGTTGGGACCACGCTTTGCTTCGCGCGCGGGCGGGTATACACGCATCTTAAAATGCGGGTTTCG
CGCGGGTGATAATGCACCTATGGCGTATATTGAATTGGTGGACCGATCGGAAAAGGCTGAGGCAGCGGCTGAATAAT
CTAGAAAGACGTC

>rpsK
ATGGCTAAAGCACCGATCCGCGCGCGCAAGCGCGTTCGTAAGCAGGTTAGTGATGGTGTTGCACACATTCACGCGTC
GTTTAATAATACAATTGTTACAATTACAGACCGCCAAGGGAATGCACTGGGATGGGCTACTGCTGGGGGGTCTGGCT
TTCGCGGCTCGCGAAAGTCTACACCTTTCGCGGCACAAGTGGCGGCGGAACGCTGTGCGGATGCGGTTAAGGAGTAT
GGGATTAAAAACTTGGAGGTGATGGTGAAGGGCCCTGGGCCTGGTCGTGAGAGCACAATCCGCGCATTAAATGCTGC
GGGATTTCGTATTACAAATATCACGGACGTTACGCCAATTCCGCACAATGGCTGCCGCCCTCCAAAGAAGCGCCGAG
TGTAATCTAGAAAGACGTC

>rpsI
ATGGCGGAGAACCAGTATTATGGTACGGGCCGTCGTAAGTCAAGTGCTGCGCGTGTGTTTATTAAGCCAGGAAATGG
CAAGATTGTGATTAATCAGCGCAGCCTAGAGCAATATTTTGGCCGCGAAACGGCTCGTATGGTTGTGCGCCAACCTT
TGGAGTTAGTGGATATGGTAGAAAAGTTAGATTTATATATTACTGTAAAGGGGGGGGGAATTTCGGGCCAAGCGGGA
GCTATTCGCCATGGGATTACACGTGCATTAATGGAGTATGATGAATCGTTGCGCAGCGAGTTACGCAAGGCAGGTTT
TGTGACGCGCGATGCACGCCAAGTGGAGCGCAAAAAGGTGGGGTTGCGCAAGGCGCGCCGCCGACCTCAATTTAGTA
AGCGCTAATCTAGAAAGACGTC

>rpsH
ATGAGTATGCAGGACCCTATTGCTGACATGTTAACGCGCATTCGCAATGGACAAGCGGCAAATAAGGCAGCTGTGAC
AATGCCGTCGTCGAAATTGAAGGTTGCTATTGCGAATGTTTTAAAAGAGGAGGGCTTCATCGAGGACTTCAAGGTGG
AGGGTGATACAAAACCGGAGTTGGAGCTAACATTAAAATACTTTCAAGGGAAGGCTGTGGTGGAGTCGATCCAACGC
GTGTCACGTCCGGGCTTACGTATTTACAAGCGCAAGGACGAATTACCTAAGGTGATGGCTGGATTGGGCATTGCGGT
AGTGAGCACAAGTAAGGGGGTGATGACGGACCGCGCTGCTCGTCAAGCGGGCCTGGGCGGTGAGATCATTTGTTATG
TTGCGTAATCTAGAAAGACGTC

>rplM
ATGAAAACATTTACTGCGAAGCCGGAAACGGTGAAGCGTGATTGGTACGTGGTGGATGCTACAGGCAAGACGCTAGG
TCGCCTAGCGACAGAGTTGGCGCGCCGTCTACGTGGGAAACATAAGGCAGAGTATACGCCACATGTGGACACTGGCG
ACTATATTATTGTGTTAAATGCGGATAAGGTGGCTGTGACTGGTAATAAACGCACAGATAAGGTATATTACCATCAT
ACGGGGCATATTGGGGGGATTAAGCAGGCAACGTTCGAGGAAATGATCGCGCGTCGCCCGGAACGCGTTATCGAGAT
TGCTGTGAAGGGTATGCTGCCGAAGGGTCCATTGGGCCGCGCGATGTTTCGCAAGTTGAAGGTGTATGCTGGGAATG
AACATAATCATGCAGCGCAACAGCCTCAGGTGCTGGATATTTAATCTAGAAAGACGTC

>rplP
ATGCTGCAGCCGAAACGAACGAAGTTTCGCAAGATGCATAAGGGACGCAATCGTGGCCTAGCTCAAGGAACTGACGT
GTCGTTTGGGTCATTTGGCTTGAAGGCGGTGGGACGAGGACGCTTGACGGCTCGCCAAATTGAGGCGGCTCGACGCG
CGATGACTCGCGCGGTGAAACGCCAGGGCAAAATTTGGATTCGCGTTTTTCCAGATAAGCCAATTACGGAGAAACCA
TTAGCGGTACGCATGGGCAAGGGCAAGGGCAATGTTGAATACTGGGTGGCGCTGATCCAACCTGGCAAGGTATTATA
CGAGATGGATGGGGTGCCAGAGGAATTAGCGCGCGAGGCGTTTAAATTAGCGGCGGCTAAGTTGCCAATCAAGACGA
CATTCGTGACGAAAACTGTTATGTAATCTAGAAAGACGTC

>rplO
ATGCGCTTGAACACATTAAGTCCTGCAGAGGGAAGCAAGAAAGCTGGCAAGCGTTTAGGCCGCGGGATTGGTAGTGG
ACTGGGCAAGACGGGCGGACGCGGCCATAAGGGCCAAAAAAGCCGTAGTGGTGGCGGAGTGCGCCGTGGATTTGAAG
GGGGGCAAATGCCGTTATATCGACGCTTACCTAAGTTTGGGTTTACGTCACGCAAGGCTGCTATTACTGCGGAGATC
CGCTTATCGGATTTAGCGAAGGTGGAGGGGGGCGTTGTGGATTTAAATACTTTGAAGGCTGCGAATATCATTGGCAT
TCAAATTGAATTTGCAAAGGTTATTTTAGCGGGAGAGGTTACTACGCCTGTGACGGTGCGAGGTTTGCGCGTGACAA
AGGGAGCGCGCGCGGCAATTGAGGCGGCGGGTGGCAAGATTGAGGAGTAATCTAGAAAGACGTC

>rplJ
ATGGCGCTAAACCTGCAGGATAAGCAGGCAATCGTGGCGGAGGTATCTGAGGTGGCGAAGGGAGCATTGAGCGCTGT
GGTGGCAGACTCGCGCGGTGTGACAGTGGACAAGATGACGGAGTTACGAAAGGCTGGCCGTGAGGCGGGGGTGTATA
TGCGCGTAGTACGCAATACGTTATTACGTCGAGCGGTGGAGGGCACACCATTTGAATGTTTAAAGGATGCATTCGTA
GGGCCAACGTTGATCGCGTATAGCATGGAGCATCCTGGAGCGGCAGCACGCTTGTTTAAGGAATTTGCTAAGGCAAA
CGCGAAGTTCGAAGTAAAGGCGGCAGCGTTCGAGGGCGAATTAATTCCTGCTAGTCAAATTGATCGTTTGGCGACAT
TACCTACGTATGAGGAGGCGATCGCGCGTCTAATGGCTACGATGAAGGAGGCAAGCGCGGGTAAGTTAGTGCGCACA
TTAGCGGCGGTGCGTGACGCTAAGGAGGCAGCGTAATCTAGAAAGACGTC

>rpsE
ATGGCTCATATTGAGAAGCAGGCGGGTGAGTTACAAGAGAAATTGATTGCAGTTAATCGTGTGAGTAAGACTGTGAA
GGGCGGCCGAATCTTTAGTTTTACTGCGTTAACAGTGGTAGGTGACGGGAATGGGCGTGTAGGGTTCGGCTATGGCA
AGGCTCGCGAGGTGCCGGCGGCTATTCAAAAGGCAATGGAGAAGGCACGCCGTAACATGATCAATGTAGCTTTGAAC
AATGGTACGTTGCAGCATCCAGTAAAGGGCGTGCATACTGGCAGTCGTGTTTTTATGCAACCAGCAAGTGAGGGCAC
AGGCATTATTGCAGGCGGCGCGATGCGTGCAGTATTGGAGGTAGCGGGAGTGCACAATGTGTTGGCGAAGGCGTACG
GCAGCACTAATCCAATTAATGTTGTGCGCGCGACGATCGACGGGTTGGAGAACATGAACTCACCGGAGATGGTTGCA
GCGAAACGCGGAAAGAGCGTGGAGGAGATCCTAGGTAAGTAATCTAGAAAGACGTC

>rplF
ATGAGTCGCGTGGCAAAGGCTCCTGTAGTGGTGCCGGCAGGTGTAGATGTGAAGATTAATGGGCAAGTGATCACTAT
TAAGGGAAAGAATGGTGAACTAACACGCACGCTGAATGACGCTGTGGAGGTAAAGCACGCTGACAACACTTTGACGT
TTGGCCCACGCGACGGATATGCTGATGGCTGGGCGCAAGCGGGGACGGCACGCGCGTTATTGAATAGCATGGTGATT
GGCGTAACAGAGGGGTTTACAAAAAAATTACAATTAGTAGGCGTGGGATATCGCGCGGCTGTGAAGGGGAACGTAAT
CAATTTGAGTTTAGGATTTTCACACCCGGTGGATCACCAATTGCCGGCAGGCATTACAGCGGAGTGCCCAACGCAAA
CGGAGATTGTATTAAAGGGGGCGGACAAACAAGTAATTGGGCAAGTGGCGGCTGACTTGCGAGCGTATCGCCGCCCG

GAACCGTACAAGGGTAAAGGCGTGCGATATGCTGATGAGGTAGTACGCACTAAGGAAGCGAAAAAAAAATAATCTAG
AAAGACGTC

>rplE
ATGGCAAAGTTACACGACTATTATAAGGATGAGGTGGTGAAGAAGCTGATGACGGAATTCAATTATAACAGCGTGAT
GCAGGTGCCACGCGTGGAAAAAATTACGTTGAATATGGGCGTGGGCGAGGCAATTGCGGATAAGAAGTTGCTAGACA
ATGCTGCGGCTGATTTGGCGGCTATTAGTGGACAGAAGCCATTAATTACGAAGGCGCGTAAGAGCGTAGCTGGTTTT
AAGATTCGCCAAGGATACCCAATTGGTTGCAAGGTGACGTTGCGCGGTGAGCGTATGTGGGAATTTTTCGAACGTTT
AATTACGATCGCGGTTCCGCGCATTCGCGATTTTCGCGGATTATCAGCGAAATCATTTGATGGGCGAGGCAATTATT
CGATGGGCGTGCGCGAACAAATTATTTTTCCGGAGATTGATTATGACAAGGTAGATCGTGTGCGCGGCCTGGACATC
ACAATCACAACGACGGCAAAGAGTGATGAGGAGGGACGTGCATTATTGGCGGCGTTCGATTTTCCATTTCGTAAATA
ATCTAGAAAGACGTC

>rpsG
ATGCCGCGCCGCCGAGTTATCGGCCAACGCAAGATCTTACCAGACCCTAAATTTGGTTCGGAGCTATTAGCGAAGTT
CGTGAACATTTTGATGGTGGACGGGAAAAAGAGTACGGCAGAGTCAATTGTGTATAGTGCATTAGAAACATTAGCGC
AACGTAGCGGCAAGTCGGAGTTGGAGGCGTTTGAGGTTGCGCTGGAGAATGTTCGACCTACGGTGGAGGTGAAAAGC
CGTCGTGTGGGCGGCAGCACGTACCAAGTGCCTGTGGAGGTGCGACCTGTACGCCGAAACGCATTAGCTATGCGATG
GATTGTGGAGGCGGCGCGCAAGCGTGGCGACAAGAGCATGGCGTTGCGTTTAGCTAATGAGCTGAGCGACGCAGCGG
AGAATAAGGGCACGGCTGTGAAAAAGCGCGAGGATGTACATCGCATGGCTGAGGCAAATAAAGCATTTGCGCATTAT
CGCTGGCTGTCTCTGCGCAGCTTCAGCCATCAAGCTGGTGCGTCATCAAAACAACCGGCGTTAGGTTATCTGAACTA
ATCTAGAAAGACGTC

>rpsD
ATGGCGCGCTACTTAGGCCCGAAACTGAAATTATCTCGCCGCGAAGGTACGGATTTGTTTCTGAAAAGTGGGGTGCG
TGCAATTGACACTAAATGCAAGATCGAGCAGGCGCCGGGTCAACATGGGGCTCGCAAGCCACGCTTAAGCGATTACG
GAGTACAACTGCGCGAGAAACAGAAGGTGCGTCGCATTTACGGAGTACTAGAACGCCAATTTCGAAATTATTATAAG
GAGGCGGCTCGCCTAAAGGGTAATACGGGCGAGAATTTACTGGCGTTGTTGGAGGGACGCTTAGATAATGTGGTGTA
TCGCATGGGGTTTGGCGCGACGCGCGCGGAGGCGCGCCAACTAGTGTCGCACAAGGCGATCATGGTGAATGGCCGAG
TAGTGAATATTGCGTCGTACCAAGTATCACCAAACGATGTGGTGTCTATCCGAGAAAAGGCTAAAAAACAAAGTCGT
GTTAAGGCGGCATTGGAACTAGCGGAACAACGCGAGAAACCGACTTGGTTAGAGGTGGACGCGGGGAAAATGGAGGG
AACTTTCAAACGCAAACCAGAACGATCAGACTTAAGCGCAGATATCAATGAGCATTTAATTGTGGAATTGTATTCAA
AATAATCTAGAAAGACGTC

>rplD
ATGGAACTGGTTCTGAAGGATGCACAAAGTGCACTAACGGTATCTGAAACGACATTTGGCCGCGACTTTAATGAGGC
ATTGGTGCATCAAGTAGTGGTGGCGTACGCTGCAGGGGCGCGCCAAGGGACACGCGCGCAAAAAACGCGCGCGGAGG
TGACAGGCTCAGGCAAGAAGCCATGGCGTCAAAAGGGTACTGGTCGAGCACGATCAGGGAGCATTAAAAGTCCAATT
TGGCGCAGCGGGGGTGTAACGTTCGCAGCGCGACCTCAAGATCATTCTCAGAAGGTAAATAAAAAAATGTATCGAGG
GGCATTAAAGTCTATTTTGTCGGAGTTAGTGCGCCAAGACCGCCTAATTGTGGTAGAAAAATTTAGCGTTGAGGCAC
CAAAGACGAAATTATTAGCGCAAAAGTTGAAGGATATGGCGCTAGAGGACGTACTAATTATTACGGGGGAATTGGAT
GAGAATTTGTTTTTAGCGGCTCGTAATTTACATAAAGTGGATGTTCGTGACGCTACGGGGATTGATCCAGTGTCATT
GATTGCGTTTGATAAGGTTGTGATGACGGCGGACGCTGTGAAACAGGTAGAAGAAATGTTGGCGTAATCTAGAAAGA
CGTC

>rplC
ATGATCGGCCTGGTTGGGAAGAAGGTAGGCATGACGCGCATTTTTACTGAGGATGGAGTGAGTATTCCGGTTACGGT
GATTGAGGTGGAGGCGAATCGTGTGACGCAAGTGAAGGATTTGGCAAATGACGGATATCGCGCAATCCAAGTTACTA
CGGGCGCGAAGAAGGCAAATCGCGTTACTAAACCGGAGGCGGGTCATTTTGCAAAGGCGGGTGTTGAGGCGGGACGA
GGGTTGTGGGAGTTTCGTTTGGCGGAGGGTGAGGAATTTACAGTTGGACAATCAATCTCGGTGGAGTTATTCGCGGA
TGTGAAGAAGGTGGATGTGACGGGTACGAGCAAGGGGAAGGGGTTTGCGGGCACAGTGAAACGTTGGAATTTTCGCA
CGCAAGATGCAACACATGGCAATTCACTGTCACATCGTGTACCTGGGAGCATTGGCCAAAATCAAACGCCAGGAAAG
GTATTTAAGGGTAAAAAGATGGCGGGACAAATGGGAAATGAGCGCGTTACAGTGCAATCGCTGGATGTGGTTCGTGT
GGATGCGGAACGTAATTTGTTGTTAGTAAAGGGCGCGGTACCAGGAGCGACGGGTAGTGATTTGATTGTGAAGCCGG
CGGTTAAAGCATAATCTAGAAAGACGTC

>rpsC

ATGGGACAAAAGGTGCACCCGAACGGCATCCGTTTGGGCATCGTGAAGCCGTGGAATTCAACTTGGTTCGCTAATAC
AAAGGAGTTTGCGGATAATTTAGATAGTGACTTCAAGGTTCGCCAATATCTAACAAAAGAGTTGGCGAAGGCTTCAG
TGAGCCGCATTGTGATTGAACGCCCAGCAAAAAGTATTCGCGTTACTATCCATACAGCACGTCCAGGGATTGTGATT
GGCAAGAAGGGAGAGGATGTGGAGAAGTTACGCAAAGTAGTGGCAGATATTGCGGGTGTGCCGGCTCAAATTAATAT
TGCTGAGGTACGAAAACCAGAGCTAGATGCGAAGTTAGTGGCGGATAGTATTACAAGCCAATTAGAGCGCCGAGTGA
TGTTTCGCCGCGCAATGAAACGCGCAGTTCAAAATGCGATGCGCTTAGGTGCGAAGGGCATCAAGGTAGAGGTATCA
GGTCGCTTAGGTGGTGCAGAGATTGCTCGAACGGAGTGGTATCGAGAGGGCCGTGTTCCTTTACATACGTTGCGAGC
AGATATTGATTATAATACGTCGGAGGCACATACGACGTATGGGGTTATTGGTGTGAAGGTATGGATTTTTAAGGGAG
AAATTTTAGGCGGCATGGCGGCTGTGGAGCAGCCTGAGAAGCCAGCAGCGCAACCAAAGAAACAACAACGCAAGGGA
CGAAAGTAATCTAGAAAGACGTC

>rpsB
ATGGCGACGGTGTCTATGCGTGATATGCTGAAAGCAGGCGTACATTTTGGCCATCAAACGCGATATTGGAATCCAAA
GATGAAACCATTTATTTTTGGCGCACGCAATAAGGTACATATTATTAATCTGGAAAAGACGGTTCCTATGTTTAATG
AGGCGTTGGCAGAGTTAAATAAAATCGCGAGCCGAAAGGGCAAGATTCTGTTTGTAGGCACAAAGCGTGCGGCGTCA
GAGGCTGTAAAGGATGCGGCGTTGTCATGTGATCAATTTTTTGTTAATCACCGTTGGTTGGGGGGCATGTTGACGAA
TTGGAAAACAGTACGCCAAAGTATTAAGCGATTGAAGGATTTGGAAACGCAAAGTCAAGATGGCACGTTTGATAAAT
TAACTAAAAAGGAGGCTCTAATGCGTACACGAGAATTGGAAAAGTTGGAGAATTCGTTGGGAGGCATTAAGGATATG
GGGGGCTTACCAGATGCATTATTCGTGATTGACGCGGATCATGAGCATATCGCGATTAAGGAGGCGAATAATTTAGG
CATCCCTGTTTTCGCGATTGTAGACACGAATAGCGACCCAGATGGCGTGGATTTTGTGATTCCAGGCAATGATGATG
CTATTCGCGCGGTTACGTTGTATTTGGGAGCGGTGGCGGCGACGGTGCGCGAGGGTCGCTCACAAGACTTAGCAAGC
CAAGCTGAGGAGTCTTTTGTTGAGGCGGAATAATCTAGAAAGACGTC

>rplB
ATGGCGGTGGTGAAGTGCAAGCCTACGTCACCTGGGCGCCGACATGTGGTAAAGGTTGTGAATCCGGAATTACATAA
AGGAAAGCCATTCGCGCCATTATTAGAGAAGAATTCTAAGAGTGGAGGCCGCAATAATAACGGGCGCATTACGACAC
GCCACATTGGAGGCGGACATAAACAAGCATATCGCATCGTAGATTTTAAGCGTAATAAGGATGGCATTCCAGCGGTA
GTGGAGCGCCTGGAATATGACCCAAATCGCTCGGCAAATATTGCTTTGGTGCTATATAAGGATGGCGAGCGACGCTA
TATTTTAGCGCCGAAGGGTTTGAAGGCGGGAGATCAAATCCAATCGGGTGTGGACGCGGCGATTAAGCCTGGAAATA
CATTGCCTATGCGTAATATTCCTGTGGGAAGCACGGTGCACAATGTTGAGATGAAGCCGGGCAAGGGGGGACAATTA
GCGCGAAGTGCGGGCACGTATGTGCAAATTGTGGCGCGCGACGGCGCGTACGTGACGTTACGCTTGCGTAGTGGCGA
GATGCGCAAGGTGGAGGCGGATTGTCGCGCGACATTGGGTGAGGTGGGTAACGCAGAACACATGTTACGTGTGTTAG
GGAAGGCTGGCGCAGCGCGTTGGCGCGGCGTGCGCCCAACTGTGCGTGGCACGGCTATGAATCCTGTGGATCATCCG
CACGGGGGCGGCGAGGGCCGCAATTTCGGCAAACATCCTGTTACGCCATGGGGTGTGCAAACGAAGGGCAAAAAAAC
TCGTTCAAATAAACGCACAGACAAGTTTATTGTGCGCCGTCGATCGAAGTAATCTAGAAAGACGTC

>prfB
ATGTTCGAGATCAACCCAGTGAACAACCGTATCCAAGATTTAACAGAGCGTAGTGATGTGCTGCGCGGATACCTAGA
TTATGATGCGAAAAAGGAACGCTTGGAGGAGGTGAATGCTGAATTAGAGCAACCTGACGTGTGGAATGAGCCGGAGC
GTGCTCAAGCACTAGGGAAGGAACGCTCGAGCCTGGAGGCGGTGGTGGATACACTGGATCAGATGAAGCAAGGTTTG
GAGGACGTGAGCGGCTTGTTGGAGTTGGCAGTTGAGGCAGATGATGAGGAGACTTTCAATGAGGCTGTGGCAGAGTT
GGATGCGTTGGAGGAGAAGTTAGCACAACTGGAATTTCGTCGCATGTTTAGCGGTGAGTACGATTCAGCAGATTGTT
ATCTGGACATCCAAGCAGGCAGCGGGGGCACAGAGGCGCAAGATTGGGCATCGATGCTGGAACGCATGTACTTACGA
TGGGCGGAGTCTCGCGGCTTTAAGACGGAGATTATTGAGGAATCTGAGGGCGAGGTAGCTGGCATCAAGAGCGTTAC
TATTAAGATTTCGGGTGACTATGCGTATGGTTGGTTGCGCACGGAAACGGGTGTGCATCGATTGGTTCGCAAGTCAC
CATTCGATAGCGGGGGGCGTCGTCATACAAGCTTTTCATCAGCTTTCGTGTACCCTGAGGTAGACGACGACATCGAC
ATTGAGATTAATCCAGCAGACTTACGTATCGATGTGTACCGTACAAGCGGAGCAGGGGGCCAACATGTGAATCGCAC
TGAGAGCGCTGTTCGCATCACTCATATTCCAACAGGTATTGTAACACAATGTCAAAATGATCGATCACAACATAAAA
ATAAGGACCAAGCTATGAAACAAATGAAGGCTAAACTGTACGAGTTAGAAATGCAAAAAAAGAACGCGGAAAAGCAA
GCAATGGAGGACAATAAGAGCGATATTGGGTGGGGATCTCAAATCCGATCGTACGTGCTGGACGATAGTCGTATCAA
GGACTTGCGTACTGGTGTTGAAACGCGTAATACTCAAGCGGTTTTAGATGGGAGTTTGGACCAGTTCATTGAGGCGT
CTTTAAAGGCGGGCCTATAATCTAGAAAGACGTC

>rpsA
ATGACAGAGTCGTTCGCACAGTTATTCGAGGAAAGCCTGAAGGAGATTGAAACGCGTCCAGGATCGATTGTGCGCGG
GGTGGTGGTGGCAATTGATAAGGATGTTGTGTTGGTGGATGCAGGCTTAAAGAGTGAAAGCGCGATTCCTGCGGAAC
AATTTAAGAATGCTCAAGGTGAATTGGAGATTCAAGTGGGAGATGAGGTGGATGTGGCGTTAGATGCGGTGGAGGAT

GGTTTTGGGGAAACGTTGTTAAGCCGCGAAAAGGCAAAGCGCCATGAGGCTTGGATTACTTTGGAGAAGGCGTATGA
GGACGCAGAGACAGTGACTGGCGTGATTAATGGAAAGGTGAAAGGGGGGTTTACGGTAGAATTAAATGGGATCCGCG
CATTTTTACCTGGCAGCTTGGTGGATGTGCGCCCAGTTCGCGATACATTGCATTTAGAGGGTAAGGAACTGGAGTTC
AAGGTGATTAAATTGGACCAAAAACGTAATAATGTGGTGGTGAGCCGCCGCGCTGTGATTGAGTCGGAGAATTCGGC
GGAACGTGACCAACTATTGGAGAATTTGCAAGAGGGTATGGAGGTGAAGGGGATTGTGAAAAATCTGACAGATTATG
GAGCTTTTGTAGACTTGGGTGGAGTGGATGGTTTGTTACATATTACGGATATGGCTTGGAAGCGTGTGAAACACCCA
TCTGAGATTGTTAATGTTGGTGATGAGATTACGGTGAAGGTTTTGAAATTTGATCGTGAGCGCACTCGCGTTTCATT
AGGGCTAAAGCAATTAGGTGAGGACCCATGGGTGGCAATTGCGAAGCGCTACCCTGAGGGAACGAAGTTAACAGGGC
GTGTTACTAATTTGACTGATTATGGTTGTTTTGTGGAGATTGAGGAGGGAGTGGAGGGATTAGTTCATGTGAGCGAG
ATGGATTGGACTAATAAGAATATTCATCCTAGTAAGGTGGTGAATGTGGGTGACGTGGTTGAGGTGATGGTGTTAGA
CATTGATGAGGAGCGCCGCCGCATTTCATTAGGATTAAAGCAATGTAAGGCGAATCCATGGCAACAATTTGCTGAAA
CGCATAATAAAGGGGATCGCGTGGAGGGCAAGATTAAATCGATTACGGATTTTGGCATTTTTATTGGACTGGATGGT
GGAATTGATGGTTTAGTGCATTTATCGGATATTTCTTGGAATGTAGCGGGTGAGGAGGCTGTGCGCGAGTATAAGAA
GGGTGATGAGATTGCAGCGGTGGTGTTGCAAGTAGATGCTGAGCGCGAGCGCATTTCTTTAGGTGTGAAGCAATTAG
CGGAGGACCCTTTTAATAATTGGGTGGCATTGAATAAAAAGGGTGCAATTGTGACTGGGAAGGTGACAGCGGTGGAT
GCGAAGGGTGCGACGGTGGAGTTAGCGGATGGGGTAGAGGGATATTTACGCGCATCGGAGGCGTCGCGCGATCGTGT
GGAGGATGCGACGTTGGTGTTGTCAGTGGGTGATGAGGTGGAGGCGAAGTTTACGGGTGTGGACCGCAAGAATCGTG
CGATTAGTTTGAGTGTGCGCGCTAAGGATGAGGCAGATGAAAAGGACGCGATTGCGACAGTAAATAAGCAAGAGGAC
GCTAATTTTAGTAATAATGCGATGGCGGAGGCATTTAAGGCGGCGAAGGGTGAATAATCTAGAAAGACGTCA

>kanR (with upstream ribosome binding site) used for transcriptional fusion with recoded genes (start codon is in lower case)
TGAGTTGTCGAGATTTTCAGGAGCTAAGGAAGCTAAAatgAGCCATATTCAACGGGAAACGTCGAGGCCGCGATTAA
ATTCCAACATGGATGCTGATTTATATGGGTATAAATGGGCTCGCGATAATGTCGGGCAATCAGGTGCGACAATCTAT
CGCTTGTATGGGAAGCCCGATGCGCCAGAGTTGTTTCTGAAACATGGCAAAGGTAGCGTTGCCAATGATGTTACAGA
TGAGATGGTCAGACTAAACTGGCTGACGGAATTTATGCCTCTTCCGACCATCAAGCATTTTATCCGTACTCCTGATG
ATGCATGGTTACTCACCACTGCGATCCCCGGAAAAACAGCATTCCAGGTATTAGAAGAATATCCTGATTCAGGTGAA
AATATTGTTGATGCGCTGGCAGTGTTCCTGCGCCGGTTGCATTCGATTCCTGTTTGTAATTGTCCTTTTAACAGCGA
TCGCGTATTTCGTCTCGCTCAGGCGCAATCACGAATGAATAACGGTTTGGTTGATGCGAGTGATTTTGATGACGAGC
GTAATGGCTGGCCTGTTGAACAAGTCTGGAAAGAAATGCATAAACTTTTGCCATTCTCACCGGATTCAGTCGTCACT
CATGGTGATTTCTCACTTGATAACCTTATTTTTGACGAGGGGAAATTAATAGGTTGTATTGATGTTGGACGAGTCGG
AATCGCAGACCGATACCAGGATCTTGCCATCCTATGGAACTGCCTCGGTGAGTTTTCTCCTTCATTACAGAAACGGC
TTTTTCAAAAATATGGTATTGATAATCCTGATATGAATAAATTGCAGTTTCATTTGATGCTCGATGAGTTTTTCTAA

Table S6-10. Refactored overlapping genes

Gene	Gene terminus overlapped	Length of overlap
rplD	C-terminus	4 bp
rplP	C-terminus	1 bp
rplW	N-terminus	4 bp
rpmC	N-terminus	1 bp
rpmC	C-terminus	1 bp
rpsQ	N-terminus	1 bp

Table S6-11. Doubling times of double mutants compared to single mutants

Gene(s)	Actual fitness[a]	Predicted fitness[b]	Actual doubling time	Predicted doubling time[c]
rplP	0.381	-	128.7	-
rpmC	0.468	-	104.7	-
rplM	0.491	-	99.7	-
rplE	0.919	-	53.3	-
rpsI	0.521	-	94.0	-
rpmC-rplM-1	-	0.230	88.0	213.0
rplE-rplM-6	-	0.452	94.2	108.4
rpmC-rpsI-4	-	0.244	86	200.9
rplP-rplM-1	-	0.187	160.8	261.9

[a]Actual fitness is the measured doubling time divided by wild type doubling time (49 minutes under the conditions used in this study)

[b]Predicted fitness is the product of the actual fitness measured for each synthetic gene corresponding to a double mutant.

[c]Predicted doubling time is the wild type doubling time (49 minutes under the conditions used in this study) divided by the predicted fitness

References:

31. S. A. Schwartz, D. R. Helinski, Purification and Characterization of Colicin E1. *J. Biol. Chem.* **246**, 6318 (October 25, 1971, 1971).
32. K. A. Datsenko, B. L. Wanner, One-step inactivation of chromosomal genes in Escherichia coli K-12 using PCR products. *Proc. Natl. Acad. Sci. U. S. A.* **97**, 6640 (Jun, 2000).
33. D. G. Yu *et al.*, An efficient recombination system for chromosome engineering in Escherichia coli. *Proc. Natl. Acad. Sci. U. S. A.* **97**, 5978 (May, 2000).
34. N. Rohland, D. Reich, Cost-effective, high-throughput DNA sequencing libraries for multiplexed target capture. *Genome Research* **22**, 939 (May, 2012).
35. S. Kosuri *et al.*, Scalable gene synthesis by selective amplification of DNA pools from high-fidelity microchips. *Nat Biotech* **28**, 1295 (2010).
36. D. G. Gibson, H. O. Smith, C. A. Hutchison, J. C. Venter, C. Merryman, Chemical synthesis of the mouse mitochondrial genome. *Nat. Methods* **7**, 901 (Nov, 2010).
37. P. A. Carr *et al.*, Enhanced multiplex genome engineering through co-operative oligonucleotide co-selection. *NAR*, 1 (2012).
38. F. J. Isaacs *et al.*, Precise Manipulation of Chromosomes in Vivo Enables Genome-Wide Codon Replacement. *Science* **333**, 348 (Jul, 2011).
39. J. A. Mosberg, C. J. Gregg, M. J. Lajoie, H. H. Wang, G. M. Church, Improving Lambda Red Genome Engineering in *Escherichia coli* via Rational Removal of Endogenous Nucleases. *PLoS One* **7**, e44638 (2012).
40. H. H. Wang *et al.*, Programming cells by multiplex genome engineering and accelerated evolution. *Nature* **460**, 894 (Aug, 2009).
41. N. R. Markham, M. Zuker, in *Methods in Molecular Biology:VOLUME II: STRUCTURE, FUNCTION AND APPLICATIONS*, J. M. Keith, Ed. (2008), vol. 453, pp. 3-31.
42. M. T. Sykes, E. Sperling, S. S. Chen, J. R. Williamson, Quantitation of the Ribosomal Protein Autoregulatory Network Using Mass Spectrometry. *Analytical Chemistry* **82**, 5038 (Jun, 2010).