A combinatorial view of canonical Wnt signaling

Citation

Permanent link
https://nrs.harvard.edu/URN-3:HUL.INSTREPOS:37365944

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
A combinatorial view of canonical Wnt signaling

Abstract

This computational thesis is concerned with two problems, one methodological, the other scientific. The scientific problem consists in understanding the behavior of the canonical Wnt signaling pathway at a molecular systems level. Several components of the pathway are scaffold proteins, capable of polymerization. This is the first model of Wnt signaling that has sought to elucidate the role of polymerization and scaffold use, notably regarding complex formation and combinatorial post-translational modifications. As for the methodological problem, I wanted to build a model based directly on a collection of mechanistic facts derived from the primary research literature, without this collection organized by some a priori understanding, or prejudice, of how the pieces fit together. The model was built in a haphazard and incremental fashion, along the way certifying its compliance with experimental observations. Its unbiased nature promoted unanticipated insights. This thesis is therefore a contribution to a new style of modeling. A technical problem consisted in identifying a platform that would permit this task. Although one such existed, it was in a testing stage. By breaking, debugging, and exploiting it many times, I contributed to its development. A further methodological problem consisted in the analysis of a model of this nature. A large model, in which behavior emerges from the asynchronous interaction of components is effectively “wild”. Building it does not yield automatic understanding of it. Insight required an approach that cannot
be described in any other term than “empirical”. Tools suitable for its analysis did not exist, and I had to build them. This also led to the making of a model’s model, in which “empirical” insights from the wild and “undomesticated” model were marshaled into a more traditional analytical one. The insights that emerged from the wild model are: (i) polymerizing scaffolds can structure larger programmable entities, enabling concentration effects through uni-molecular interactions; (ii) the β-catenin destruction complex is not a classical complex with fixed stoichiometry, but a statistical machine; (iii) the Wnt signalosome is not an entity distinct from the destruction complex it antagonizes; rather one complex literally morphs into the other, never completely abolishing destruction activity.
# Contents

1. **The Kappa platform**  
2. **The Trace Query Language**  
3. **Other Projects**  
   3.1 Kappa Snapshot Analyzer  
      3.1.1 Introduction  
      3.1.2 Core classes  
      3.1.3 Scripts and functions  
   3.2 Kappa Code Highlighter  
      3.2.1 Writing a decent lexer  
      3.2.2 Having the lexer do something  
      3.2.3 Making the *something* be useful  
4. **Combinatorial protein-protein interactions on a polymerizing scaffold**  
4.1 Main Text  
4.2 Supplemental Text  
5. **Genesis and consequences of large complexes in Wnt signaling**  
5.1 Introduction  
5.1.1 Wnt signaling challenges tools, formalisms, and conceptions  
5.1.2 The contents of the model  
5.1.3 This is not “the one true model” of canonical Wnt signaling  
5.2 The assembly of protein aggregates in the Wnt system absent β-catenin  
5.2.1 Assembly of the destruction state prior to Wnt addition  
5.2.2 Wnt mediated effects  
5.3 Outlook  
6. **Activities of large complexes in Wnt signaling**  
6.1 On resizing a stochastic simulation  
6.2 Recapitulation of empirical observations  
6.3 Multiple steady states accessible to a down-sized model  
6.3.1 Outlook  
6.4 Causal study of the events leading to degradation
6.4.1 Conclusion .......................................................... 166
6.5 Discussion .............................................................. 168

7 ON VARYING AFFINITIES, ABUNDANCES, AND NUMBER OF BINDING SITES 170
  7.1 Polymerization of Axin through cross-polymerization with APC can trigger a phase
transition into a de-mixed state ........................................... 171
    7.1.1 Motivation ....................................................... 171
    7.1.2 Methodology .................................................... 172
    7.1.3 Results .......................................................... 173
    7.1.4 Discussion ....................................................... 182
  7.2 Size and composition of the destruction complex exhibits sensitivity to protein levels
and robustness to affinity strength ....................................... 184
    7.2.1 Motivation ....................................................... 184
    7.2.2 Methodology .................................................... 184
    7.2.3 Results .......................................................... 185
    7.2.4 Discussion ....................................................... 195
  7.3 Take home messages ................................................ 196

8 Conclusion ............................................................... 197

APPENDIX A PRESENTATION OF THE Wnt MODEL 200
  A.1 Canonical Wnt ...................................................... 202
    A.1.1 Destruction Complex interactions ............................ 203
    A.1.2 Signalosome interactions ..................................... 227
    A.1.3 Other interactions ............................................. 233
  A.2 Non-canonical Wnt ................................................ 249

APPENDIX B ALGEBRA AND SOME DERIVATIONS 254
  B.1 The two body problem .............................................. 254

APPENDIX C DEFINITION OF THE PATTERNS USED AS OBSERVABLES 258

REFERENCES ............................................................... 272
To my husband, Zack.
Acknowledgments

• My collaborators, the talented computer scientist Jonathan Laurent and the outstanding
  mathemagitian Andrés Ortíz.

• The original Kappa five, who created and continue to shepherd the development of the for-
  malism: Jean Krivine, Jérôme Féret, Russ Harmer, Vincent Danos, and Walter Fontana.

• Past members of the lab, who have helped me mature as a scientist, an academic, and (slightly)
  as a programmer: Pierre Boutillier, Mutaamba Masha, Winston Anthony, Ioana Cristescu,
  Sebastian Légaré, Tomáš Mikula, Xing Li, and Mitchell Newberry.

• The department of Systems Biology, for hosting an environment where this type of theoreti-
  cal work can be done.

• The program in Systems, Synthetic, and Quantitative Biology (née Systems Biology), which
  allowed me to structure my graduate education as I saw fit.

• Friends outside Academia who help in anchoring and retaining perspective, Clint Cavanaugh
  and Doug Quade.

• My parents, Elva and Héctor Sr, who discouraged an aversion to complexity and long tasks,
  and encouraged my scientific development.

• My husband, Zack Schilling, and extended family Linda and Tim, for providing support and
  encouragement, and lasagna.

• The Harvard University Graduate Student union, which has provided the most rewarding
  networking across disciplines I’ve experienced at Harvard. Sitting on the Bargaining Com-
  mittee for the students, negotiating with University administrators, has exposed me to ar-
  guments and postures difficult to encounter outside the negotiation room. And there have
  been few moments as satisfying as seeing an article I wrote reach the state of Tentative Agree-
  ment, providing a financial benefit to over four-thousand people. There is so much more to
  Academia than what happens in a laboratory.
• And last but not least, my advisor, Walter Fontana. “A PhD should be a protected experience, where the student has the freedom to explore topics and make mistakes”; I doubt this project would have been possible in any other group. I had the freedom to follow my research wherever it led, and that allowed us to stumble onto behavior no body anticipated.
Summary of contributions

A major focus of my graduate work has been the construction and analysis of the largest, most detailed computational model of canonical Wnt signaling to date (chapters 5 to 7). I was in particular aiming at studying the combinatorial aspects of the pathway; this required using a tailor-made modeling language, known as Kappa, which is based on graph re-write rules in the style of chemistry. In the process of building this model, I have participated in the development, testing, and debugging of the Kappa platform (chapters 1 and 2). Likewise, I developed novel tools to analyze and render Kappa models (chapter 3), specifically the snapshot analyzer (section 3.1) and the Kappa code highlighter (section 3.2). The insights from the computational model allowed me to derive an analytically tractable abstraction, i.e. a “model-of-a-model” (chapter 4), that provided a deeper understanding of the observations in the larger model.
As the primary power-user of the Kappa platform, my contributions lie in pushing the development of the tools to work with large models. For this article, my data contributed Figures 5 & 6, their discussion, and the design of the visualization mode used for them (i.e. contact maps as circular maps, snapshots as patchworks).
For the contact map  A model with close to a hundred sites imposes the need to both visualize all sites, as well as maintain their readability. Traditional layout engines, based on simple networks, are unable to handle this complexity—a simple graph layout yields unreadable hairballs, whereas techniques designed for big data applications obscure too much detail. This tension required a new layout technique, and we converged to using a circular layout engine for the contact maps in this integrated development environment.

For the snapshots  As for snapshots, I had encountered the patchwork layout when producing the figures for my undergraduate thesis. Of the several layout and visualization engines available under the Graphviz package, the patchwork layout managed to display the complexity of snapshots while preserving the readability of mass distribution. Through my experience explaining what mass distributions look like for snapshots, we incorporated the three ways of weighting patch sizes, by complex mass, by complex size, and by complex abundance. Through the years, the concept of patchwork layouts has re-emerged as tree-layouts.
The Kappa platform for rule-based modeling

Pierre Boutillier1,*, Mutaamba Maasha1, Xing Li1,2, Héctor F. Medina-Abarca1, Jean Krivine3, Jérôme Feret4, Ioana Cristescu1, Angus G. Forbes5,* and Walter Fontana1,*

1Department of Systems Biology, Harvard Medical School, Boston, MA 02115, USA, 2Edgewise Networks, Burlington, MA 01803, USA, 3IRIF, Université Paris-Diderot – Paris 7 F-75205 Paris, France, 4Département d’informatique de l’ENS (INRIA/ENS/CNRS), PSL Research University, F-75005 Paris, France and 5Department of Computational Media, UC Santa Cruz, Santa Cruz, CA 95064, USA

*To whom correspondence should be addressed.

Abstract

Motivation: We present an overview of the Kappa platform, an integrated suite of analysis and visualization techniques for building and interactively exploring rule-based models. The main components of the platform are the Kappa Simulator, the Kappa Static Analyzer and the Kappa Story Extractor. In addition to these components, we describe the Kappa User Interface, which includes a range of interactive visualization tools for rule-based models needed to make sense of the complexity of biological systems. We argue that, in this approach, modeling is akin to programming and can likewise benefit from an integrated development environment. Our platform is a step in this direction.

Results: We discuss details about the computation and rendering of static, dynamic, and causal views of a model, which include the contact map (CM), snapshots at different resolutions, the dynamic influence network (DIN) and causal compression. We provide use cases illustrating how these concepts generate insight. Specifically, we show how the CM and snapshots provide information about systems capable of polymerization, such as Wnt signaling. A well-understood model of the KaiABC oscillator, translated into Kappa from the literature, is deployed to demonstrate the DIN and its use in understanding systems dynamics. Finally, we discuss how pathways might be discovered or recovered from a rule-based model by means of causal compression, as exemplified for early events in EGF signaling.

Availability and implementation: The Kappa platform is available via the project website at kappa-language.org. All components of the platform are open source and freely available through the authors’ code repositories.

Contact: pierre.boutillier@ens-lyon.org or angus@ucsc.edu or walter_fontana@hms.harvard.edu

1 Introduction

Statistical analysis and visualization efforts have become indispensable for interpreting and navigating the swell of data produced by rapid advances in high-throughput methods at the single-cell level. The abundance of mRNA transcripts, the localization of proteins and their post-translational modifications are data taken to reflect the state of a biological system. The overwhelming effort at analysis and visualization to date has been directed at data originating from such sweeping surveys of system state.

Meanwhile, detailed mechanistic studies are elucidating the structural and post-translational requirements on protein regions, domains and residues that enable specific interactions. These data do not directly pertain to system state, but to the processes that generate system state. The many interactions inferred from biochemical, biophysical and structural deep drills have been combined into static networks. Surveying the properties of such networks, while useful, offers limited insight, since the significance of any one interaction is determined by the dynamic behavior of all interactions that co-occur in a given situation. Likewise, static depictions of pathways are narratives that might serve to organize data, yet pathways do not exist as physical circuits like road networks do; rather, at any moment, pathways emerge from and are maintained by the many concurrent and changing interactions between the molecular agents that populate a system.

Mechanistic models are needed for understanding systems dynamics and making interventions into cellular processes more deliberate. Yet, such models are often viewed with suspicion, because much of the mechanistic detail is missing, either for lack of
knowledge or the need to curtail complexity (or both). The utility of mechanistic models appears further diminished by statistical models that can yield prediction without concomitant understanding. Another issue is the perception that mechanistic models, because difficult to build, are rarely kept in sync with a rapidly evolving knowledge base. These are not arguments against the need for mechanistic models in interpreting interaction data. Rather, these arguments articulate the need for mechanistic models that are scalable, easy to update and fork and based on a formal foundation conducive to computer-aided reasoning. In this contribution, we lay out our ideas and their implementation in support of this vision.

2 The rule-based approach

Technology for making, running and analyzing large dynamic models, though in its infancy, is progressing significantly (Bachman and Sorger, 2011; Cohen, 2015; Gyori et al., 2017; Leow and Schaff, 2001). One powerful component are rule-based languages, such as Kappa (Danos et al., 2007a) and BioNetGen (Faeder et al., 2009; Harris et al., 2016) for molecular biology and Mod (Andersen et al., 2016) for organic chemistry.

Common to rule-based languages are entities with a structure represented as a graph and rules that are graph-rewrite directives (Fig. 1a and b). The point of a rule is to distinguish between the transformation of a structure fragment and the reaction instance resulting from it in the context of specific entities. This distinction is a key organizing principle in chemistry (Fig. 1a). Since a particular interaction between proteins often appears to depend on some but not all aspects of their state, rule-based languages adapt the chemical perspective to molecular systems biology by viewing proteins as higher-order atoms and non-covalent associations between proteins as higher-order molecules (Fig. 1b).

A rule-based language sets a specific level of granularity at which rules ‘axiomatize’ interactions. For example, a rule of chemical transformation, as in Figure 1a, only exposes the net result of underlying electronic rearrangements, which are governed in turn by ‘arrow (or electron) pushing’ rules (Kermack and Robinson, 1922) at a lower level of abstraction. Although not explicitly represented, these processes are not ignored, as they inform what a rule should say. Likewise, rules of protein interaction, Figure 1b and c, are based on structural considerations, bioinformatic sequence analysis and biochemical mechanisms reported in the literature (or hypothesized by the modeler). Yet, a rule does not expose these lower-level aspects, but summarizes their overall effect in terms of pre- and post-conditions on protein state.

2.1 Agents, patterns, embeddings and activity

At the heart of languages like Kappa and BioNetGen lies the agent abstraction, Figure 1c, which conceptualizes a protein as an agent with an interface of sites that represent distinct interaction capabilities, such as binding and post-translational modification (Fig. 1c). Through their sites, agents can connect into site graphs (Fig. 1). A site graph that exhibits the full interface and state of all its agents. A rule r: L → R, involves two site graphs, L and R, which usually mention some but not all sites of their agents. L and R are therefore patterns, not molecular species (Fig. 1).

The state of a system is itself a graph consisting of a (large) ensemble of disconnected site graphs, each representing one instance of a molecular species. We call such an ensemble a mixture (as in reaction mixture) and denote it by M. A rule r is applied to a mixture M by embedding L into M, which means a match in M of all agent types, site names and states (including binding states) mentioned in L. A rule is executed by replacing the part of the mixture matched to L with R (Fig. 2).

A model is a collection of rules with an initial mixture. The stochastic behavior of a model is explored using continuous time Monte–Carlo (Boutillier et al., 2017a; Danos et al., 2007b; Gillespie, 2007; Sneddon et al., 2011) or CTMC for short. For this purpose, a rule r is assigned a constant, $\gamma_r$, which is the instantaneous probability rate that the rule triggers on any given embedding of L in M (Fig. 2). The activity $\alpha_r$ of a rule r (i.e. its propensity to fire) depends on the total number of embeddings of L in M (mass action), denoted by $|\mathcal{L}|\mathcal{M}$, and is given by $\alpha_r = \gamma_r|\mathcal{L}|\mathcal{M}/\sigma_r$, where $\sigma_r$ is the number of symmetries of L preserved by r. The term $\|\mathcal{L}|\mathcal{M}/\sigma_r$ is the number of physical configurations that are
distinguishable with respect to the mechanism expressed in \( r \). The activity \( ar \) is the stochastic rule-based analog of the reaction in standard chemical kinetics. If a reaction event occurs at time \( t \), then the probability that \( r \) is applied to one of its embeddings is given by its relative activity \( \frac{ar}{\sum ar} \).

Even when the set of all reachable molecular species and their reactions can be deduced from a given collection of rules, the size of the resulting network is often too large to be handled explicitly. Importantly, translating rules into kinetic differential equations based on an explicit reaction network would miss the point of rule-based approaches entirely. Rules provide compactness, transparency and a handle on combinatorial complexity; and, perhaps most significantly, systems of rules constitute a more appropriate level for causal analysis than reaction networks, because reasoning at the level of rules avoids contamination with context that defines a reaction but is irrelevant to its application (Figs 1b and 2).

### 2.2 Syntax

From here on our discussion refers specifically to the Kappa framework (https://kappalanguage.org). Graphs are the proper formal objects in this framework, but we need a textual notation for convenience. The binding state of a site \( s \) is specified in square brackets: \([\cdot]\) indicates that \( s \) is free (unbound), whereas \([s] \) (\( n \) a non-negative integer) is bound to the unique other site in the same expression whose binding state is also \( n \). Thus, \( A[x[2]], B[y[2]] \) asserts that agent \( A \) is bound at \( x \) to agent \( B \) at \( y \). Given an agent type with two binding sites, \( A(x, y) \), a simple rule might be, \( \text{bind}: A[x, [], y] \rightarrow A[x[1], y[1]] \). This rule yields 'head-to-tail' polymerization, because it is matched by any two asymmetric ends (sites \( x \) and \( y \)). Rule \( \text{bind} \), therefore, generates linear polymers and rings of any size, limited only by the available pool of agents of type \( A \). A site \( z \) can also have an internal state, expressed as a string between curly braces. The internal state is typically used to specify a post-translational modification. Thus, \( C[z[p][\cdot]] \) denotes an agent of type \( C \) whose site \( z \) is in state \( p \) (phosphorylated, say) and unbound. When writing patterns, such as observables or rules, states not mentioned do not constrain their matchings into the mixture (‘don’t care, don’t write’). For example, \( C[z[p][\cdot]] \) is matched by any instance of \( C \) phosphorylated at \( z \) regardless of its binding state. An extensive manual (Boutillier et al., 2017b) provides many details beyond the present scope.

### 3 Materials and methods

#### 3.1 Kappa software agents

The main Kappa ecology consists of several software agents that communicate through an ad hoc JSON-based protocol and expose high-level functionalities through both a Python API and an HTTP REST service. At present these are: The Kappa Simulator (KaSim), the Kappa Static Analyzer (KaSA) and the Kappa Story Extractor (KaSTOR). A Python client enables scripting to tailor workflows and is available as the kappy package in pip. Modeling in a rule-based language is much like writing large and complex programs, which necessitates an integrated development environment. To integrate the new Kappa web services, we developed a browser-based User Interface (UI), which also comes packaged as a self-contained application for the Windows OS, Mac OS and Ubuntu. This effort raised several challenges pertaining to visualization and analysis that we address below.

The KaSim is an implementation of CTMC designed for rule-based models with particular emphasis on scalability in typical usage regimes (Danos et al., 2007b). A recent optimization targets a costly step in the CTMC core loop (Boutillier et al., 2017a). After a rule has been applied to the mixture, rule activities (Section 2.1) must be updated by identifying all embeddings that were gained or lost (Fig. 2) and likewise for updating counts of user-defined observables. The optimization consists in the initial construction of a model-dependent data structure by virtue of which no pattern—whether the left side of a rule, a user-defined observable, or any sub-pattern of these—is ever scanned twice for the same embedding into the mixture. For example, if two left sides \( L_c \) and \( L_d \) have a common sub-pattern, any match to that sub-pattern is shared during the attempt at extending the match to \( L_c \) and \( L_d \). KaSim is interactive, allowing a simulation to be paused and resumed at any time. A versatile intervention language can be used for defining perturbations, such as altering rate constants and adding/removing agents interactively or at time points set in the input file. The intervention language supports queries about the state of the system and requests for data dumps, such as snapshots of the mixture and rule influence matrices (Section 3.3).

Simulation is the main tool for studying the dynamical behavior of a rule-based model. However, some properties of a model can be determined by direct inspection and manipulation of the rules without execution. Such properties are static in the sense of being time-independent, yet they relate to behavior because they demarcate its influence (positive, negative, or none) as between states at two sites of the same agent (i.e. assertions of the kind ‘whenever site \( s \) is phosphorylated, site \( p \) is bound to agent \( D \)’). The firing of a rule \( r \) could create a configuration that partially satisfies the \( L_r \) of another rule \( s \), thus potentially contributing to an increase in the activity of \( s \). Similarly, \( r \) might compete with \( s \) for instance, thus potentially decreasing the activity of \( s \). Such potential influence (positive, negative, or none) is also a static property.
In Section 3.3, we look at the actual influence that rules have on each other, which is a dynamic property. Static properties are especially helpful in debugging a model, as they can quickly pinpoint discrepancies between expected and actual behavior. The KaSA supports all these analyses. It relies heavily on a technique called ‘abstract interpretation’, described in Section 3.2.

A third tool is the KaSTOR which aims at providing insights into how an event of interest (EOI) occurred in a particular event trace obtained by simulation of a model. The underlying concepts offer a distinctive perspective on (mechanistic) causality in molecular systems. We outline their implementation in Section 3.4.

3.2 Static analysis and abstract interpretation

The formal analysis of graph-rewrite systems relies heavily on elucidating the relationships between graph-morphisms, for which category theory provides a powerful framework. Yet, in practice, computing an exact answer to problems like enumerating the set of reachable species is not feasible, because of numbers beyond imagination. Static analysis algorithms written for Kappa achieve practical utility by deploying an approach known as abstract interpretation (Cousot and Cousot, 1977), wherein the given rules are made more abstract in the sense of systematically discarding specific information. The reachability problem is then approached within the set of abstract rules, where it can be handled faster at the price of an over-approximated rather than exact solution.

Central to the abstract interpretation technique in Kappa is the concept of a local view. The local view of an agent A that occurs in a site graph G consists of A itself, its sites and their state as mentioned in G, alongside information about which sites of A are bound to which site of which agent type, i.e. without including the bound agent and any additional information about it. For example, in Figure 3, the local view of the first agent (labelled ‘1’) of type A is given by ii, which consists of A, site d bound to site d of another agent of type B (but without including that agent, hence the dotted outline) and site s bound to site s of an agent of type B. We can think of such linkage information as a bond ‘stub’, expressed syntactically as A[s[\(s_B\)], d[d[A]]]. The local view of the other A-agent (labelled ‘2’) is the same. Thus, the complex i yields a set of three distinct local views (ii, iii, iv). This step is called abstraction, denoted by (\(\approx\)). Continuing the example, we can stitch the local views back together using the information in the bond ‘stubs’. Doing this in all possible ways yields the set of four complexes shown in Figure 3. This step is called concretization. Clearly, the abstraction step for agent A loses information about the phosphorylation state of the B bound to it. The concretization step reconstructs the original complex, but also others—an overapproximation. We apply this idea to the left side \(L\), and right sides \(R\), of each rule \(r\) in a model. In this way, \(L \rightarrow R\) is replaced by an abstract version \(\approx(L) \rightarrow \approx(R)\) cast entirely in terms of local views. It turns out that the fixed point obtained from the forward closure of abstract rules always exists and is finite (even for systems capable of generating infinitely many species). It can be further shown that stitching together these local views in all admissible ways yields a superset of the true reachable molecular species (Danos et al., 2008). Because of overapproximation, a species \(Q\) in the superset obtained from abstraction may not be reachable with the original rules. However, if \(Q\) is not in the superset, it cannot be reached by the original rules either: A no is a no, and a yes is maybe. The most recent version of KaSA generalizes local views to more flexibly compromise between accuracy, efficiency and expressivity (Feret and Ly, 2018).

This technique makes reachability analysis fast enough to permit computing, thus visualizing, contact maps (CMs; Section 4.1) and rule influence maps in real-time as a user updates a model. This in turn makes abstract reachability useful in debugging a model, which is a process that furthers understanding.

As a simple example of KaSA’s operation consider the following minimal model snippet. (Rate constants are ignored by static analysis; they only serve the purpose of getting the model past the parser).

\[
\begin{align*}
\%init: 1 E () \\
\%init: 1 R () \\
\end{align*}
\]

\[
\begin{align*}
\text{‘r1’ } & E(\{.\}, R(\{.\}) -> E(\{.\}), R(x[1]) @0 \\
\text{‘r2’ } & E(\{.\}, R(x[1], c[.]) -> E(\{.\}), R(x[1], c[.]) @0 \\
\text{‘r3’ } & R(x[2], c[.], c[.], R(x[1], x[.], c[.]) @0 \\
\text{‘r4’ } & R(c[1], c[.], n[.], R(c[1], c[.], n[.]) @0 \\
\text{‘r5’ } & R(c[1], c[.], n[.], R(c[1], c[.], n[.]) -> R(c[1], c[.], n[.]) @0 \\
\text{‘r6’ } & R(c[1], c[.], n[.], R(c[1], c[.], n[.]) @0 \\
\text{‘q1’ } & R(x[.], c[.], c[.], n[.]) @0 \\
\text{‘q2’ } & R(c[1], c[.], c[.], n[.]) -> R(c[1], c[.], c[.], c[.], n[.]) @0 \\
\text{‘q3’ } & R(c[1], c[.], c[.], n[.]) -> R(c[1], c[.], c[.], c[.], n[.]) @0 \\
\end{align*}
\]

This snippet is a simplified version of reversible receptor/ligand binding (‘r1’, ‘r2’), receptor dimerization (‘r3’, ‘r4’) and internal crosslinking (‘r5’, ‘r6’). (The underscore denotes a bound state without caring about the type of binding partner.) To query the static analyzer about the reachability of patterns of interest, we defined two identity rules, ‘q1’ and ‘q2’ in which \(L_{q1} = R_{q1}\), to exploit KaSA’s detection of dead rules. Dead rules are rules that can’t fire because their left side pattern is unreachable. As soon as ‘q1’ and (or) ‘q2’ are added to ‘r1’−‘r6’ in the UI editor, KaSA flags them as unreachable. For ‘q1’ this means that a receptor cannot be cross-linked unless it is bound to ligand (but no single rule explicitly states that); for ‘q2’ this means that the bonds at sites c and cr of a receptor must both engage the same second receptor, i.e. no receptor triples. KaSA came to this conclusion by automatic abstract interpretation. The analysis made use of the initialization statements (‘init’) for ligand and receptor R. (Static analysis only cares whether the initial copy number is non-zero.) If we remove the first statement, KaSA responds in real-time by flagging all rules as unreachable—because of the absence of ligand. This can be useful to ensure that the initial condition of complex models covers the intended agent types. The command line version of KaSA provides many additional analyses that are not yet exposed in the software agent.

3.3 The dynamic influence network

The dynamic influence network (DIN) is a construct for tracking the way rules influence each other over time. Its visualization is discussed in Section 4.3.
A rule $r$ fires with a probability proportional to its activity $x_r$ (Section 2.1). When $r$ fires, it alters the state of the mixture $M$, which can affect the activity of rule $s$ by generating or destroying matchings to $C_s$ in $M$. Fix a time interval $[t, t + \tau]$ and let $i$ index the events that occur in that interval. The $i^{th}$ event causes a transition from state $M_i$ (before the event) to $M_{i+1}$ (after the event). If the $i^{th}$ event is due to the firing of $r$, its contribution to the relative change in the activity of $s$, denoted by $\Delta x_{rs}(i)$, is given by

$$\Delta x_{rs}(i) = \begin{cases} \frac{x_r(i+1) - x_r(i)}{x_r(i)} & \text{if } i \text{ is due to } r \text{ and } x_r(i) \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

(1)

We aggregate these contributions during the interval $[t, t + \tau]$ and divide by the number $n_r$ of $r$-events to obtain the average influence of $r$ on $s$ during that interval:

$$\overline{\Delta x}_{rs}(t, \tau) = \frac{1}{n_r} \sum_{i \in [t, t+\tau]} \Delta x_{rs}(i),$$

with the events $i$ occurring in $[t, t + \tau]$. We refer to the matrix Equation (2) as the DIN. The nodes of the DIN are rules and an event $i$, and the $r$-events to obtain the average influence of $r$ on $s$ during that interval:

3.4 Causal lineages and compression

KaSim can output the whole sequence of events generated by a simulation. This is called a trace. The traces of models we work with comprise routinely many millions of events. A persistent challenge is to extract an explanation of how an EOI specified by the user actually occurred in a particular run.

Given a trace $\tau$, i.e. a sequence of events $\tau = e_1 e_2 \cdots e_n$ to an EOI (EOI $\equiv e_n$), we first identify which events were on the path to the EOI and in which causal order, as the temporal sequence of events may not always be relevant in stochastic systems with concurrency. Two temporally adjacent events $e_1 e_2$ are concurrent if they can occur in any order $e_2 e_1$ or $e_1 e_2$—while having the same effect, otherwise there is a relation of precedence between them. For $e_2$ to precede $e_1$ means that the rule underlying $e_2$ contributes to satisfying a condition required by the left-hand side of the rule underlying $e_1$. The naive approach we take here is to view causality as the opposite of concurrency, i.e. a relation of precedence. Not all precedence is causal, however. Suppose a kinase can phosphorylate a substrate and, independently, the substrate can be degraded. Whenever phosphorylation is observed, it must precede degradation, but, given our assumption, we cannot say that phosphorylation caused degradation.

We now wish to identify the events in $\tau$ that were involved in bringing about the EOI while revealing their precedence relations. Figure 4 illustrates the method by means of a simple example. Assume the EOI is the production of a free phosphorylated substrate in the following Michaelis–Menten type model of:

\[
\begin{align*}
'b' & \rightarrow K[z[1]], S[x[1], y(u)] \\
'w' & \rightarrow K[z[1]], S[x[1]] \\
'p' & \rightarrow K[z[1]], S[x[1], y(p)] \\
'*' & \rightarrow S[x[1], y(p)] \\
*d' & \rightarrow S[x[1], y(u)] \\
'
\end{align*}
\]

Since causality is a relation between events, the identity rule `*' serves to transform our query into an event. Suppose a particular mixture contains three substrate agents $S$ and two kinase agents $K$ and that a simulation happened to produce the trace $\tau = bbubpuppus$, where each event is labeled by the rule that produced it and contains information about which agent identifiers matched the rule. KaSTOR, the causal analysis software agent, attaches to every site of every agent instance in the mixture a thread and records whether an event only tested the state of a site or whether it also modified it. In typical Kappa rules, a modification also implies a test. This results in the representation depicted in Figure 4a, where threads run vertically and events horizontally with markings—black filled circles if the thread was modified, white if it was only tested. To reconstruct the causal past of `*' or `w', we slide its tests backward in time following the instructions shown in Figure 4b. Consider an event $e$ followed by event $e'$ and ask for every site involved in $e'$ whether the test or modification of that site could have occurred before $e$. If this is the case, slide that test or modification backwards in time past $e$ and repeat that question against the event that occurred prior to $e$, etc. A test of the state at a particular site in event $e'$ can always slide past a test (or absence thereof) for the same site in event $e$, as it will not invalidate the occurrence of $e$. A test associated with event $e'$ cannot slide past a modification associated with event $e$, or $e'$ could not have occurred. This generates a causal arrow from $e$ to $e'$. A modification cannot slide past a modification for the same reason. Lastly, a modification can slide past a test, but this is a case of non-causal precedence discussed above and is annotated by adding a ‘prevention arrow’ from $e'$ to $e$.

This procedure, applied to our trace, will discard any events that had no bearing on the EOI (Fig. 4d), while producing a directed acyclic graph representing the precedence structure of its causal past (Fig. 4c). Figure 4c reads: $S_2$ binds $K_1$, then they dissociate, then $K_1$ binds $S_1$, then they dissociate; then $K_1$ rebinds $S_2$, then $K_1$ phosphorylates $S_2$ and precedes (non-causally) the dissociation of $K_1$ from $S_2$; thus causing the EOI. The problem is that events in the causal past of an EOI often violate necessity. In our example, the detour...
through \( S_1 \), although it occurred, was not necessary, since event \( e_7 \) could have happened instead of \( e_1 \) yielding the same result. One way to isolate necessity is to search, within the causal past reconstructed by the previous procedure, for a minimal sub-set of events that can produce the EOI (Danos et al., 2012). We call this minimization causal compression. KaSTOR automatically translates a causal past into a propositional formula in which each event is associated with a Boolean variable whose value determines whether the event is kept or discarded. Each scenario has implications for which other events must be retained or discarded to ensure that the EOI obtains. KaSTOR then seeks the smallest number of propositional variables that, when set to true (keep), satisfy the formula. This is done with KaSTOR’s own SAT solver tailored to the structure of this problem. The result is the compressed causal past of Figure 4e, which we call a story. A more realistic case of causal compression is touched upon in Section 4.4 and Figure 8.

4 Results

The Kappa UI is a suite of interactive visualization tools, integrating various services in the Kappa ecology. Here, we provide details about the Kappa CM, the snapshots tool, the DIN visualization (DIN-Viz) and our initial efforts in visualizing stories created with KaSTOR.

A range of projects introduce techniques for visualizing protein–protein interaction networks in support of analysis (Murray et al., 2017). Kohn et al. (2006) introduce a technique that visualizes regulatory networks, but is not rule-based in the sense defined here. Chylek et al. (2011) provide guidelines for visualizing and annotating rule-based models extending the notion of CM (Danos et al., 2007a). Smith et al. (2012) develop the RuleBender framework for editing and exploring rule-based systems. This is taken further by Sekar et al. (2017), in particular by grouping and merging rules into a more compact diagrammatic representation. Dang et al. (2015) present a visualization technique that uses animation to highlight causal relationships within pathways, and Paduano and Forbes (2015) explore interactive methods to visualize common patterns in biological networks.

4.1 Visualization: the CM

The CM of a model is a generalized site graph in which every agent type occurs exactly once and every site exhibits all bonds and states that are possible according to the rules. The CM is generated automatically. For smaller to moderately sized models, its construction and visualization are updated in real-time while a user is writing rules in the UI text editor. The CM allows a user to quickly check visually whether a molecular species is admissible in principle. The static analyzer KaSA can constrain the CM based on abstract reachability, as outlined in Section 3.2. The most immediate utility of such visualization is in debugging. For example, a missing arc, where a modeler is expected to see one, might indicate an omitted, mistyped, or unreachable rule.

Figure 5a depicts the CM of a large model as rendered in the UI. By inspection of the CM one can deduce the existence of proper cycles. A cycle is a path in the CM that starts at an agent type, visits other agent types by following bonds and returns to the original type. A proper cycle is a cycle in which the departure site at any agent differs from the landing site. A proper cycle indicates the potential for polymerization, which is a significant property as it implies an infinite set of possible molecular species. The enumeration of proper cycles in larger CMs, such as in Figure 5, requires computational assistance provided by KaSA.

Scalable visualization is a challenge. In this instance, we partially address scalability through interactivity by providing a zooming feature and the automatic toggling of detail (information resolution) along with selective emphasis as the user hovers with the mouse over elements of the CM. In Figure 5b, for example, the mouse hovers over Axin (red), highlighting its interconnections with other agent types of the model.

4.2 Visualization: snapshots

Simulations are often discussed in terms of time series tracking the abundance of molecular species. On their own, such time series are less useful when dealing with combinatorial models, because it is impractical to track all possible molecular species and it is often unclear which of them are salient. This is where visualizing snapshots of the system state at different levels of resolution is useful. A snapshot is a view of the mixture, i.e. the state of the system, at a given time. Presently, snapshots have three levels of resolution, two of which are currently available through the UI. The first level is a ‘patchwork’ (treemap) rendering of the mixture and comes in three views (Fig. 6a–c): count, size and mass. In all views, the term patch refers to a rectangle separated from others by white space. A patch represents a molecular species, but visualizes it only in terms of agent composition, akin to a sum formula in chemistry. The relative abundance of an agent type within a species is shown as a monochromatic rectangle within the patch. Two monochromatic patches of the same color show up separately if the underlying species differ in size (e.g. monomer versus n-mer) or are equal in size but differ in the state of at least one agent. These distinctions are made explicit by clicking on the patch, which opens the next higher level of resolution—connectivity—for that species (Fig. 6d). All three views at the first level contain the same number of patches, but the area of a patch depends on the view. In the count view, the area reflects the relative abundance of the molecular species represented by the patch; in the size view it reflects the size of the species, i.e. the number of connected agents and in the mass view the area is determined by count times size. The differences between these views are particularly notable in systems with polymerization, such as a model of canonical Wnt signaling whose CM is shown in Figure 5. In this
system, several scaffold proteins can form a polymeric structure, either alone or in interaction with each other (Fiedler et al., 2011).

We illustrate the patchwork view by comparing the state of the system before (−) to that after (+) Wnt addition. In both cases the most frequent components in the count view are homogenous, mainly protomers or dimers of scaffolds like Dvl, APC and LRP6. The size view, however, is dominated by a large composite polymer—so large that it also dominates the mass view. Prior to Wnt addition, the polymer has the composition of a so-called 'destruction complex', which targets β-catenin (CTNNB1) for degradation. The before/after comparison in the size view yields the following observations. (i) The largest complex has increased in size. (ii) The largest complex has changed composition, gaining a large mass of Dvl (dark brown), previously spread across a variety of separate entities. (iii) The rest of the mixture has become far more fragmented by a greater diversity of smaller species. (iv) The diversity of LRP6 (dark pink) states has increased—prior to Wnt we recognize a few patches in the lower right corner of the size view, while there are many more monochromatic LRP6 patches after Wnt addition. This is more conspicuous in the count view. (v) Proteins that were associated with Dvl prior to Wnt addition (especially Frz) have now been pulled into the giant component, representing the migration of the destruction complex machinery to the membrane where it interacts with trans-membrane proteins, such as Frz and CK1d. (vi) The giant component from the pre-Wnt state has lost a large amount of APC, which is now found in isolated association with β-catenin. (vii) Complexes with a composition of more than four agent types are more frequent. Prior to Wnt addition, the only complex with more than four agent types was the destruction aggregate, whereas afterwards the giant component increased in compositional diversity and so did species not connected to it.

Such observations are drawn more readily from patchwork diagrams than time series, especially if it is unclear what to look for at the outset. Patchwork diagrams lay out an overall view of a suitably coarse-grained system state and seem especially conducive for qualitative comparisons. The observations made here, regardless of their biological significance, also illustrate the complexity of phenomena that can arise with a mere dozen agent types—a complexity that would be difficult to capture without a rule-based platform with an integrated visualization environment.

A higher level of informational resolution is accessed when clicking on patches to reveal the connectivity structure of the underlying molecular species, as shown for the pre-Wnt case in Figure 6d. The polymeric cross-linked structure dominating the pre-Wnt regime is shown on the left of that panel, alongside with the second largest structure, revealing a completely segregated aggregate of Dvl. At this resolution, the structures do not exhibit (internal) state and site information, which would not be cognitively scalable. The next level of resolution consists in revealing detailed (local) state information in a split view for any agent that the user hovers over in the connectivity view. An interactively accessible resolution hierarchy increases the effectiveness of the patchwork view.

4.3 Visualization: the DIN
KaSim can be directed to compute influence matrices $\Delta I_{r,s}(t, \tau)$, Equation (2), aggregated over user-specified time intervals $[t, t + \tau]$. Typically, $\tau$ and increments in $\tau$ are chosen so as to create overlapping intervals for smoothness. A time series of such matrices can be uploaded to the DIN visualization server (DIN-Viz) (https://github.com/CreativeCodingLab/DynamicInfluenceNetworks), where the DIN is presented as a node-link diagram, leveraging a force-directed layout to position related nodes and clusters near to each other. The nodes represent the rules and the links convey the influence of rules on each other, color-coded with red (green) signifying a negative (positive) influence. This visualization enables the topological analysis of rule-fluxes in a KaSim simulation at the chosen time step scale. By mapping the magnitude of influence to the link strength, highly mutually influential rules are closely grouped together visually. This provides insight as to which rules might (temporarily) conspire in producing a pathway.

In order to further facilitate the analysis of these highly related rules, we perform a clustering operation on the network based on the influence between rules to generate groups with a high inner influence, based on a user-selected threshold. The absolute value of influence between rules, determines the attractive link forces in the network. DIN-Viz maps the number of firings of a rule (during
[t, t + 1] to the node size, and the influence of one rule on another to the link width. We indicate the directionality of a link and the sign of the influence by using directional color gradients. Figure 7 uses yellow-to-red for negative and yellow-to-green for positive; colorblind-safe colormaps are also available.

Details-on-demand are available for each link, showing the source and target, as well as the exact influence value; similarly, hovering over a rule (node) shows the rule name, the amount of self-influence and the rule’s top incoming and outgoing influences. The user may also choose to make visible the names of the rules as labels, either for all rules, or for interactively selected rules.

To represent the dynamism of the system, animation is used to update the influence between the rules, which in turn updates the edge weights, node sizes and cluster definitions. A time slider controls the current time step, enabling the user to move through time or jump to a particular time step. Standard playback controls animate the simulation so that the user can observe changes in the DIN over time.

Although force-directed layouts mitigate visual clutter in node-link diagrams, dense networks can still be difficult to make sense of—an issue that is exacerbated when representing large datasets. DIN-Viz enables the user to manually create a layout of nodes or entire clusters through relocating and ‘pinning’ them to specified locations. This reorganization reduces clutter, but also helps users to distribute rules and clusters in a way that is cohesive with their thought process during exploratory analysis. When pinned, the spatial positioning and grouping of selected rules and clusters is preserved over the course of the entire animation, over-riding the normal layout behavior.

While the mathematically defined clusters capture groups of rules that influence one another, there may be rules which are related through their behavior but lack a strong influence with one another. To solve this problem, we implement a ‘painting’ interaction. The user can provide a color marking to nodes to indicate that they are grouped together, and then insert them into an existing or newly created cluster. These categorical groupings, similar to the spatial groupings achieved through pinning, aid in the logical organization of rules during analysis by the user.

As a use case we analyze a rule-based model of the autonomous KaiABC circadian oscillator in the cyanobacterium Synechococcus elongatus. We forgo a detailed description of the molecular biology and of the Kappa model, which closely follows the literature (van Zon et al., 2007), in favor of a broader outline illustrating the reasoning enabled by the DIN and its visualization.

The KaiABC system consists of three proteins, A, B and C. C can be phosphorylated and dephosphorylated at multiple sites, thereby assuming distinct phosphorylation levels (p-levels for brevity). It also can switch between two conformational states, A and I. At low p-levels C prefers the A-state. The probability of a flip from A to I increases with increasing p-level. When C is in the A-state, it binds A with an affinity that decreases rapidly with increasing p-level. Upon binding A, C gets locked into the A-state, which promotes phosphorylation. As the p-level increases, A dissociates from C, allowing it to flip into the I-state, which favors dephosphorylation. This process results in a p-level oscillation of individual C molecules. Since these oscillations are not coordinated, no p-level oscillations will occur at the macroscopic level of the C-population. Co-ordination between C proteins is achieved by B, which binds C in the I-form. Once bound, B locks C into the I-form, facilitating dephosphorylation. Crucially, once bound to C, B also binds A with an affinity that is maximal at intermediate p-levels of C. By sequestering A in a mechanism that depends on C molecules that are late in the cycle, B holds back the phosphorylation of C molecules that are ahead in the cycle, statistically synchronizing the individual cycles and resulting
in oscillations at the macroscopic population level (van Zon et al., 2007), as seen in Figure 7a.

Panels 7b and 7c summarize the insights from the animated DIN by focusing on two snapshots taken at the midpoints of the downward and upward leg of the macroscopic phosphorylation cycle (Fig. 7a). A detailed account of the animation and its interpretation is provided elsewhere (Forbes et al., 2018). The point to note here is the drastic difference in the influence structure of the system between the two time points. This difference characterizes the synchronization mechanism described above. The structure in the upper part of the influence network consists of three groups of activity. Group (i) is a single rule responsible for dissociating A from B; group (ii), also a single rule, is responsible for the binding of A to C in the A-state and group (iii) is a family of rules that control the binding of A to B at various p-levels of the C agent to which B is bound. At t = 58, the structure (i, ii, iii) is disconnected from the lower part of the DIN, which consists of several rule families that control state flipping of C (group iv), binding of B to C (v) and background phosphorylation and dephosphorylation (vi). The structure (i, ii, iii) shows that A agents, once liberated from B, tend to be reabsorbed by available B-C complexes, rather than binding to C-agents in the A-state. This situation puts the breaks on those C-agents that are ready to initiate the upward leg in their cycle. Once the lagging C-cohort has caught up, A becomes again available to bind C and facilitate its phosphorylation. This in turn raises the activity of rule families (vii) and (viii), which control various mechanisms of the dissociation of A from C. Families (vii) and (viii) were absent from the DIN at t = 58. In the thick of this new phase, t = 67, most A agents that dissociate from C can rebind other C agents in the A-state, resulting in the feeder stream from groups (vii) and (viii) toward (ii). In other words, group (iii), no longer captures free A agents. The structure (i, ii, iii) is effectively a systemic break pedal. It is pressed in the downward phase and released in the upward phase. This organizational switch is clearly recognizable and interpretable in the DIN-viz animation.

4.4 Causal analysis

As a proof-of-concept for causal compression of traces (Section 3.4), we consider a small rule-based model capturing early events in EGF signaling. The model consists of agents representing ligand EGF, receptor EGFR, Shc, Grb2, Sos and rules that capture well-known mechanisms of interaction between these proteins. The EOI is the recruitment of Sos. A typical run involves about 30 000 agents and generates traces of 0.5 M events with many occurrences of the EOI. The causal past of the first EOI in one of these traces is shown in Figure 8 and was obtained with the procedure described in Section 3.4. The subsequent causal compression with KaSA’s SAT solver succeeds in removing the many cyclical event sequences that return the system to equivalent states with respect to the EOI. The outcome is a compact interpretable and formal pathway fragment. In this case, the compressed story describes EGF binding, subsequent receptor dimerization and autophosphorylation, which is a pre-condition for the recruitment of Shc and its phosphorylation. Independent of the EGF-induced lineage, Sos binds Grb, which, again independently, binds the Shc that was phosphorylated along the EGF lineage. The two binding events, which can happen in any order, come together in completing the recruitment of Sos.

This story pertains only to one particular EOI in one particular trace. Other EOI instances and traces might compress to other stories. One challenge, therefore, is the aggregation of story statistics over long traces or large samples of short traces to determine the most salient pathways to the EOI. Rule-based modeling gathers mechanistic information about local interactions without preconceptions as to what constitutes a pathway; rather, pathways emerge dynamically from these interactions and our automated approach detects them.

5 Conclusion

In situations not vitiated by complexity, an assumed understanding of some key aspect of a system typically precedes modeling. In the case of complex interaction networks, however, an initial understanding at the systems level may not be readily available even with reasonable knowledge of local interactions. This leads to an inversion—from understanding precedes modeling to modeling precedes understanding—that significantly alters the character of models and the practice of modeling. The traditional criteria that establish a good model—insight, elegance, conciseness, conceptual fertility—are no longer available right away. Rather, a good model should be a data structure that constitutes a transparent, editable, formal and executable representation of the facts it rests upon. This is a prescription for replacing a world we don’t understand with a model we don’t understand but that is easier to analyze and experiment with. The challenge is to develop mathematical techniques and a sound software infrastructure for analyzing, visualizing, manipulating, simplifying—in short, reasoning with—models that are like empirical objects.

We see two major challenges that rule-based approaches need to address. The first consists in developing formal and computational techniques for constructing explanations. The glimpse on causality in Sections 3.4 and 4.4 is just a beginning. Second, languages like Kappa and BioNetGen are not knowledge representations. Tying modeling to knowledge representation requires a structured staging area designed to formally organize the abstraction process leading from biochemically rich and grounded descriptions to logical, ungrounded, but executable expressions (Basso-Blandin et al., 2016; Harmer et al., 2017). Modeling occurs, to a large extent, at that intermediate staging level. In addressing challenges like these, there will be ample need for innovative visualization.

Funding

This work was sponsored by the Defense Advanced Research Projects Agency (DARPA) Big Mechanism Program and the US Army Research Office under grant numbers W911NF-14-1-0367 and W911NF-14-1-0395.

Conflict of Interest: none declared.

References


The Trace Query Language

Kappa was designed with causality in mind. Obtaining the full chain of events that yielded a particular event of interest (EOI) is straightforward in Kappa, and is what we call a trace.

In a model with millions of proteins, the trace often is a chain of millions of events. We can reduce

* Whereas λ-calculus represents logic, and π-calculus represents process, κ stands for kausality and komplexity. The name came into being long before I joined the group.
the complexity by focusing exclusively on the events directly on the causal past of our EOI, removing events that did not contribute to it, and grouping concurrent events on the same tier. This yields a partial-order graph, culminating in the EOI and starting at the simulation’s initialization, what we call a story. However, in a model with phosphatases and kinases, loaded on the same complex, the do-undo loop generates a large number of serial events all contributing, all causal, to the EOI; in parallel, polymerization and scaffold use inserted large assembly processes in the chain of events. To tackle large stories, the Kappa community had developed heuristics for story compression. These were effective when applied to the small models the community had, like EGF signaling or DNA damage repair; for Wnt, they yielded trivial and obvious answers.

Jonathan Laurent (CMU) and I collaborated on a completely novel approach at causality over traces. Motivated by the Wnt use-case, the trace query language focuses on events, rather than molecular species or even rules. Instead of measuring the concentration of something over time, we look at events and their causal links. One can answer questions like “is the size of the complex, measured in Axins, the same for the event of hyperphosphorylation of β-catenin as it is for the subsequent hyperphosphorylations, ubiquitinations, and the degradation event?” The trace query language allows us to focus on the events experienced by an agent and the relationships between such events at the single-agent resolution, rather than on measures of global states (i.e. snapshots) or abundances over time (i.e. traces).
A Trace Query Language for Rule-Based Models

Jonathan Laurent\textsuperscript{1,2}, Hector F. Medina-Abarca\textsuperscript{2}, Pierre Boutillier\textsuperscript{2}, Jean Yang\textsuperscript{1}, and Walter Fontana\textsuperscript{2}

\textsuperscript{1} Carnegie Mellon University, Pittsburgh, USA
\texttt{jonathan.laurent@cs.cmu.edu}
\textsuperscript{2} Harvard Medical School, Boston, USA

Abstract. In this paper, we introduce a unified approach for querying simulation traces of rule-based models about the statistical behavior of individual agents. In our approach, a query consists in a trace pattern along with an expression that depends on the variables captured by this pattern. On a given trace, it evaluates to the multiset of all values of the expression for every possible matching of the pattern. We illustrate our proposed query language on a simple example, and then discuss its semantics and implementation for the Kappa language. Finally, we provide a detailed use case where we analyze the dynamics of β-catenin degradation in Wnt signaling from an agent-centric perspective.

Keywords: Rule-based modeling \cdot Query language \cdot Kappa

1 Introduction

Rule-based modeling languages such as Kappa \cite{4} and BioNetGen \cite{9} can be used to write mechanistic models of complex reaction systems. Models in these languages consist of stochastic graph-rewriting rules that are equipped with rate constants indicating their propensity to apply. Together with an initial mixture graph, these rules constitute a dynamical system that can be simulated using Gillespie’s algorithm \cite{2,5,8}. Each run of simulation results in a sequence of transitions that we call a trace.

In practice, simulation traces are often discarded in favor of a limited number of global features, such as the concentration curves of a set of observables. However, a more detailed analysis of their structure and statistical properties can provide useful insights into a system’s dynamics. For example, causal analysis methods exist \cite{4,6} that compress a large trace into a minimal subset of events that are necessary and jointly sufficient to replicate an outcome of interest, and then highlight causal influences between those remaining events. Queries about the statistical behavior of individual agents can lead to complementary insights. Examples include (i) measuring the average lifespan of a complex under different conditions, (ii) computing a probability distribution over the states in which a
particular type of agent can be when targeted by a given rule, and (iii) estimating how much of a certain kind of substrate getting phosphorylated is due to a particular pathway at different points in time.

In this paper, we propose a unifying language to express queries of this kind, that are concerned with statistical features of groups of molecular events that are related in specific motifs. These motifs are formalized using a notion of trace pattern. Then, evaluating a query comes down to computing the value of an expression for every matching of a pattern into a trace. We give a first illustration of this paradigm on a toy example in Sect. 2. After that, we introduce our query language in Sect. 3 and give it a formal semantics. We then characterize a natural subset of this language for which an efficient evaluation algorithm exists and discuss our implementation for the Kappa language (Sect. 4). Finally, we leverage our query engine to explore aspects of the dynamics of the Wnt signaling pathway in a detailed use case (Sect. 5).

2 A Starting Example

In order to illustrate our Trace Query Language, we introduce a toy Kappa model in Fig. 1. It is described using a rule notation that has been introduced in the latest release of the Kappa simulator and which we borrow in our query language. With this notation, a rule is described as a pattern that is annotated with rewriting instructions. The pattern denotes a precondition that is required for a rule to target a collection of agents. Rewriting instructions are specified by arrows that indicate the new state of a site after transformation.

The model of Fig. 1 features two types of agents: substrates $S$ and kinases $K$. Both kinds of agents have two different sites, named $x$ and $d$. In addition, $x$-sites can be in two different internal states: unphosphorylated and phosphorylated. We write those states $u$ and $p$, respectively. Rule $b$ expresses the fact that a substrate and a kinase with free $d$-sites can bind at rate $\lambda_b$. Rules $u$ and $u^*$ express the fact that the breaking of the resulting complex happens at different rates, depending on the phosphorylation state of the kinase involved. Finally, rule $p$ expresses the fact that a substrate that is bound to a kinase can get phosphorylated at rate $\lambda_p$. In all our examples, we consider initial mixtures featuring free substrates and kinases in similar quantity. Substrates are initially unphosphorylated and kinases are present in both phosphorylation states.

By playing with this model a bit, one may notice that the concentration of phosphorylated substrate reaches its maximal value faster when the ratio of phosphorylated kinases is high (given the rules of our model, the latter quantity is invariant during the simulation). This phenomenon cannot be explained by looking at rule $p$ alone. The query provided in (1) can be run to estimate the probability that a substrate is bound to a phosphorylated kinase when it gets phosphorylated:

$$\text{match } t : \{ S \left( x \xrightarrow{u \rightarrow p}, \, d^1 \right), \, k : K \left( d^1 \right) \} \quad \text{return } \text{int state} [ \bullet t ] ( k, "x")$$  (1)

Given a trace, this query matches every transition where a substrate is getting phosphorylated and outputs the phosphorylation state of the attached kinase.
Fig. 1. An example Kappa model. On the left, it is described using the edit notation introduced in KaSim 4. Numbers in a rule expression correspond to local bond identifiers and \( \bullet \) indicates a free site. Sites not mentioned in a rule are left unchanged by it. A graphical representation is provided on the right. Phosphorylated sites are indicated in grey. Dotted and solid arrows indicate slow and fast reactions, respectively.

The variables \( t \) and \( k \) denote a transition and an agent, respectively. Moreover, the expression \( \text{int.state} \[ \bullet t \] ( k, "x") \) refers to the internal state of the site of agent \( k \) with name "x" in the mixture preceding transition \( t \).

Running the previous query, we learn that substrates are much more likely to be phosphorylated by kinases that are phosphorylated themselves, even when such kinases are in minority in the mixture. This leads us to conjecture a causal link between the phosphorylation state of a kinase and its efficiency. After some thoughts, this link can be easily interpreted: because \( \lambda_u \gg \lambda_u^* \), phosphorylated kinases form more stable complexes with substrates, leaving more chances for a phosphorylation interaction to happen. In fact, the average lifespan of a kinase-substrate complex is exactly \( \lambda_u^{-1} \) when the kinase is phosphorylated and \( \lambda_u^{-1} \) when it is not. We can check these numbers experimentally by running the following query:

\[
\text{match } b : \{ s:S( d^* \rightarrow 1), \ K( d^* \rightarrow 1, \ x_p ) \} \\
\text{and } \text{first } u : \{ s:S( d \rightarrow *) \} \text{ after } b \\
\text{return } \text{time}[ u ] - \text{time}[ b ]
\]

This query outputs a multiset of numbers, whose mean is the average lifespan of a complex formed by a substrate and a phosphorylated kinase. The same quantity can be computed for unphosphorylated kinases by replacing \( x_p \) by \( x_u \) in the first line of (2). The pattern in this query does not match single transitions but pairs of related transitions \( (b, u) \), where \( b \) is a binding transition and \( u \) the first unbinding transition to target the same substrate.

More generally, a query is defined by a pattern \( P[t, a] \) and an expression \( E[t, a] \), which feature a shared set \( t \) of transition variables and a shared set \( a \) of agent variables. The pattern \( P \) can be regarded as a predicate that takes as its arguments a trace \( \tau \) and a matching \( \phi \) mapping the variables in \( t \) and \( a \) to actual transitions and events in \( \tau \). The expression \( E \) can be regarded as a function that
maps such \((\tau, \phi)\) pairs to values. Then, the query evaluates on a trace \(\tau\) to the multiset of all values of \(E\), for every matching \(\phi\) that satisfies \(P\) in \(\tau\).

3 The Core Query Language

In this section, we introduce the extensible core of our proposed query language and give it a formal semantics.

3.1 Meaning and Structure of Queries

As shown in Fig. 2, a query \(Q\) consists in a pattern \(P\) and an expression \(E\). It can be interpreted as a function \(\llbracket Q \rrbracket\) from traces to multisets\(^1\) of values. The set of allowed values can grow larger as richer expressions are added to the language. Our current implementation defines a value as a tuple of base values and features the following types for base values: bool, int, float, string, agent, agent_set and snapshot.

A pattern \(P\) is interpreted as a function \(\llbracket P \rrbracket\) that maps a trace to a set of matchings. A matching \(\phi\) is defined by two functions \(\phi_t\) and \(\phi_a\), which map variable names to transition identifiers and agent identifiers, respectively. We call \(\phi_t\) a transition matching and \(\phi_a\) an agent matching. Given a trace \(\tau\) and a matching \(\phi\), the transition variable \(v\) denotes the transition \(\tau[\phi_t(v)]\), where \(\tau[i]\) is a notation for the \(i^{th}\) transition of a trace. In addition, an expression \(E\) is interpreted as a function \(\llbracket E \rrbracket\) that maps a pair of a trace and a matching to a value. The expression language is extensible and is discussed in Sect. 3.3. Its syntax is documented in Fig. 3. Then, the semantics of a query can be formally defined as follows:

\[
\llbracket \text{match } P \text{ return } E \rrbracket(\tau) = \{ \llbracket E \rrbracket(\tau, \phi) \mid \phi \in \llbracket P \rrbracket(\tau) \}.
\]

Our language constraints the structure of possible patterns. As shown in Fig. 2, a pattern consists in a sequence of clauses, which can take one of three different forms: \((t : T)\), \((\text{first } t : T \text{ after } t')\) and \((\text{last } t : T \text{ before } t')\). Here, \(t\) and \(t'\) are transition variables and \(T\) is a transition pattern. In all three cases, we say that \(t\) is constrained by the clause.

3.2 Transition Patterns

A transition pattern can be thought as a predicate that takes as its argument a pair \((t, \phi_a)\) of a transition and an agent matching. Our current implementation supports specifying transition patterns using KaSim’s edit notation. Transition patterns defined this way are enclosed within curly brackets. For example, in query (1) of Sect. 2,

\[
\{ S \left( x_{u \rightarrow p}, \ d^1 \right), \ k : K \left( d^1 \right) \}
\]

\(^1\)Note that multisets are indicated in Fig. 2 using Dijkstra’s bag notation, whereas sets are indicated using the standard curly brackets notation.
is true for a transition \( t \) and a matching \( \phi_a \) if and only if \( t \) has the effect of phosphorylating a substrate that is bound to the kinase with identifier \( \phi_a(k) \). Formally, a transition pattern \( T \) is interpreted as a function \( [T] \) that maps transitions into sets of agent matchings. Using the predicate terminology, one may say that \( \phi_a \in [T](t) \) if and only if \( (t, \phi_a) \) satisfies \( T \).

Our query language can be instantiated with any choice of a language specifying transition patterns. The only requirement is that transition patterns should be decidable efficiently in the following sense. Given a transition pattern \( T \) and a transition \( t \), one should be able to efficiently compute whether \( [T](t) \) is empty and generate an element of it in the case it is not. Our evaluation algorithm relies on this property.

### 3.3 Expression Language

We show in Fig. 3 the syntax of our expression language. An expression can consist of an agent variable, a constant, a parenthesized expression, a binary operation, a function\(^2\) of an expression, a tuple of expressions or a measure.

Measures are the basic constructs through which information is extracted from a trace. They come in two different kinds: state measures and event measures. State measures are used to extract information about the state of the mixture at different points in the trace. They are parametered with state expressions that can take the form \( \cdot t \) or \( t \cdot \), denoting the states before and after transition \( t \), respectively. For example, the \( \text{int\_state} \) measure that is used in (1) is a state measure. In addition, event measures are used to extract information that is about a transition itself (in contrast to the states that it connects). They are parametered by transition variables. For example, the \( \text{time} \) measure that is used in (2) is an event measure.

The expression language can be easily extended with new operators, functions, measures and types. In the same way than the language for specifying transition patterns, it should be regarded as a parameter of our query language and not as a rigid component.

### 4 Evaluating Queries

In this section, we introduce a natural subset of the language described in Sect. 3, for which we provide an efficient implementation. Queries in this subset are said to be regular, and they display an interesting rigidity property.

#### 4.1 Rigidity

Intuitively, a pattern is said to be rigid if its matchings are completely determined by the value of a single transition variable.

\(^2\) Note that functions always take a single argument, which can be a tuple.
query \( Q \) := match \( P \) return \( E \) & [\( Q \) \in Trace \Rightarrow \{ Value \}]

pattern \( P \) := \( C \) & [\( P \) \in Trace \Rightarrow \{ Matching \}]

| \( C \) and \( C \)

clause \( C \) := \( t : T \) & [\( C \) \in Trace \Rightarrow \{ Matching \}]

| first \( t : T \) after \( t' \)
| last \( t : T \) before \( t' \)

transition pat. \( T \) := \( \cdots \) & [\( T \) \in Transition \Rightarrow \{ Matching \}]

expression \( E \) := \( \cdots \) & [\( E \) \in Trace \times Matching \Rightarrow Value]

\[
\begin{align*}
[\text{match } P \text{ return } E](\tau) &= \{\phi \in [P](\tau) \mid \phi \in E\} \\
[\text{C and } C'](\tau) &= [C](\tau) \cap [C'](\tau) \\
[t : T](\tau) &= \{\phi \mid \phi \in [T](\tau[\phi_t(t)])\} \\
[\text{first } t : T \text{ after } t'](\tau) &= \{\phi \mid \phi \in [T](\tau[\phi_t(t)]) \land \phi_t(t') < \phi_t(t) \land \forall i. \phi_t(t') < i < \phi_t(t) \Rightarrow \phi \notin [T](\tau[i])\} \\
[\text{last } t : T \text{ before } t'](\tau) &= \{\phi \mid \phi \in [T](\tau[\phi_t(t)]) \land \phi_t(t) < \phi_t(t') \land \forall i. \phi_t(t) < i < \phi_t(t') \Rightarrow \phi \notin [T](\tau[i])\}
\end{align*}
\]

Fig. 2. Syntax and semantics of the trace query language

expression \( E \) := \( a \mid C \mid (E) \mid E \bowtie E \mid f(E) \mid E \square E \mid M_s[S] \mid M_s[S](E) \mid M_s[t] \mid M_s[t](E) \)

constant \( C \) := \( 0 \mid 1 \mid \cdots \mid "foo" \mid \cdots \)

binary operator \( \bowtie \) := \( + \mid - \mid = \mid < \mid \cdots \)

function \( f \) := \( \text{agent_id} \mid \text{size} \mid \text{count} \mid \cdots \)

state measure \( M_s \) := \( \text{int\_state} \mid \text{component} \mid \text{snapshot} \mid \cdots \)

state expression \( S \) := \( *t \mid t* \)

event measure \( M_e \) := \( \text{time} \mid \text{rule} \mid \cdots \)

(with \( a \) an agent variable and \( t \) a transition variable)

Fig. 3. Syntax of expressions
Definition 1. Given a Kappa model, a pattern $P$ is said to be rigid if and only if it features a transition variable $r$ called root variable such that for any trace $\tau$ that is valid in the model, we have

$$\forall \phi, \phi' \in \llbracket P \rrbracket(\tau), \phi_t(r) = \phi'_t(r) \implies \phi = \phi'.$$

For example, the pattern $P$ of query (2) is rigid, with root variable $b$. Indeed, suppose that $b$ is matched to a specific transition $t$. Then, the agent variable $s$ is determined by $t$ as no more than one substrate can get bound during a single transition given the rules of our model (Fig. 1). Finally, $u$ is uniquely determined as the first unbinding event that targets $s$ after $b$.

An easy consequence of Definition 1 is that the number of matchings of a rigid pattern into a trace is bounded by the size of this trace.

4.2 Regular Queries

Our evaluation algorithm handles a subset of queries whose patterns admit a certain tree structure. For those patterns, rigidity is implied by a weaker notion of local rigidity.

Definition 2. Given a Kappa model, a transition pattern $T$ is said to be rigid if and only if for any agent variable $a$ that appears in $T$ and every valid transition $t$, we have

$$\forall \phi_a, \phi'_a \in \llbracket T \rrbracket(t), \phi'_a(a) = \phi'_a(a).$$

Intuitively, a transition pattern is rigid if matching it to a transition determines all its agent variables.

Definition 3. Given a model, a pattern $P$ is said to be locally rigid if it features only rigid transition patterns. Then, a transition variable $t$ is said to determine an agent variable $a$ if there is a clause of $P$ that constrains $t$ and features $a$.

For patterns with a particular structure, local rigidity implies rigidity. This structural assumption can be expressed in terms of a pattern’s dependency graph.

Definition 4. The dependency graph of a pattern $P$ is a graph whose nodes are the transition variables of $P$ and for which there is an edge from $t$ to $t'$ if and only if $P$ contains a clause of the form $(\text{first } t' : T \text{ after } t)$ or $(\text{last } t' : T \text{ before } t)$.

We can now define the notion of a regular pattern, and thus of a regular query.

Definition 5. A pattern is said to be regular if the following three conditions hold: (i) it is locally rigid (ii) its dependency graph is a tree (iii) whenever two of its transition variables determine a same agent variable, one of them has to be a descendent of the other in the dependency tree.

\footnote{As defined in Sect. 3.1.}
This structure enables an efficient enumeration of the matchings of a regular pattern into a trace. Moreover, the number of these matchings is bounded by the size of the trace, as regular patterns can be proven to be rigid.

**Proposition 1.** Regular patterns are rigid.

Finally, regular queries are defined as expected.

**Definition 6.** A query is said to be regular if its pattern is regular.

This notion of regularity may appear unintuitive at first, and we agree that its formal definition is somewhat involved. However, we argue that regular queries are exactly those queries that admit a natural operational interpretation. Therefore, experimentalists tend to think in terms of regular queries instinctively.

### 4.3 Evaluating Regular Queries Efficiently

When designing an algorithm for evaluating trace queries, one has to keep in mind that the corresponding sequence of state mixtures cannot fit in random-access memory all at once, even for small traces. In fact, even the most economic representation of a trace, which is specified by an initial mixture and a sequence of labeled rewriting events, may fail to fit in memory in some cases. Therefore, as often as possible, one should only be allowed to stream such a representation from disk, recomputing intermediate states dynamically and never keeping more than a small number of them in memory at once (two in our case).

Our algorithm for evaluating a regular query proceeds in two steps. First, it streams the trace to compute the set of all matchings of the pattern. Then, it streams the trace a second time to compute the value of the expression for all these matchings. The second step is quite simple to implement. Indeed, once the matchings are known, it is easy to compute the sequence of all measures that need to be performed and order them in increasing order of time. The first step attempts to match the root variable of the pattern to every transition in the trace. For each candidate matching, it uses rigidity to determine all other variables progressively as the trace is streamed, in an order that is determined by the dependency tree and with a minimal amount of caching. Overall, the algorithm runs in linear time with the length of the trace.

### 4.4 Our Implementation

We provide an implementation of our proposed trace query language, which relies on the algorithm that is mentioned in Sect. 4.3 for evaluating regular queries. Our query engine takes for inputs (i) a file that contains a list of queries written in the same syntax that we use in our examples and (ii) a trace file that has been generated by the Kappa simulator using the `trace` option. It evaluates all queries at once and generates one output file per query, in comma-separated values (CSV) format.\(^4\)

\(^4\) Every line of an output file represents a single value. In our expression language, values are tuples of base values. These are separated by commas within a line.
Queries that are non-regular for structural reasons – i.e. that fail to meet criterion (ii) or (iii) of Definition 5 – are rejected immediately. As there is no easy static check for local rigidity, queries that do not meet this criterion are rejected at runtime.

We now introduce a use case in which we leverage our query engine to explore aspects of the dynamics of the Wnt signaling pathway.

5 A Use Case on Wnt Signaling

In this use case, we are focusing on a simplified model of the β-catenin destruction complex from canonical Wnt signaling. This complex is highly conserved in animals, and operates from humans to nematodes to insects to amphibians, regulating the establishment of the dorso-ventral axis. It is also heavily involved in colon cancer.

A source of complexity in our model is the fact that none of the enzymes involved in destroying β-catenin bind it directly. Instead they are loaded onto a scaffold. Moreover, the scaffold can head-to-tail homopolymerize, in addition to having three independent binding sites on a second scaffold, itself capable of dimerization. This allows a complex of scaffolds, where connection paths or stoichiometries are dynamic. It is this complex that acts as a super-scaffold to bring the substrate in contact with the enzymes. Considering both scaffolds contain large regions of disorder (i.e. chunks of unfolded peptide with high flexibility), it is sensible to believe an enzyme loaded on one scaffold could modify the substrate loaded on the neighboring scaffold. Lacking experimental evidence to suggest a ballpark limit for this reachable horizon, we leave it unconstrained: an enzyme will be able to modify any substrate loaded onto its complex.

Another source of complexity is that having kinases (i.e. enzymes that add a phosphate group) and phosphatases (i.e. enzymes that remove a phosphate group) loaded on the same complex will result in unimolecular do-undo loops. Conceivably the kinetics of complexes will vary heavily with the amount of kinases, phosphatases, and substrates loaded onto them.

We leverage our trace query engine to explore the dynamics of this system. More precisely, we develop queries to probe the agent-centric dephosphorylation dynamics, to measure the time it takes for an agent to navigate the modification steps, and to explore the complexes at which events happen.

Our results are relevant to other pathways in addition to Wnt, from NFκB to RAS/ERK to the most studied protein on the world, P53; the pathways these proteins regulate make heavy use of polymeric scaffold complexes, sequential modifications, and do-undo loops.

5.1 Experimental Protocol and Queries

To explore our system, we create a Kappa model with three parametrizations. The model contains the scaffold proteins Axin1 (Axn) and APC, the kinases CK1α (CK1) and GSK3β (GSK), the protein phosphatases PP1 and PP2, and
the substrate of all these reactions, β-catenin (Cat). The destruction complex recruits Cat through Axn. It then gets phosphorylated at the Serine on position 45 (S45) by CK1. While S45-phosphorylated, it can be phosphorylated at the Threonine on position 41 (T41) by GSK. While T41-phosphorylated, it can be phosphorylated on both Serines on positions 37 and 33 (S37 and S33). Once S37- and S33-phosphorylated, Cat is degraded. Meanwhile, PP1 undoes the phosphorylations of CK1, while PP2 undoes those of GSK. Each kinase-phosphatase pair also compete against each other for binding sites on Axn.

The three parametrizations explore the relationship between phosphatase/kinase ratio and the distribution of do-undo events. The three parameter pairs are 50/10, 10/10, and 10/50, all in units of number of agents, and represent the number of kinases and phosphatases in the model (e.g. 10/50 presents 10 copies of PP1, 10 copies of PP2, 50 copies of CK1, and 50 copies of GSK). The scaffolds remain at an abundance of 100 each. The models begin with an initial amount of Cat of 500 agents, and the models are run for 500 simulated seconds. We use global stochastic rates for our reactions, a bi-molecular binding of $10^{-4}$ per second per agent, a uni-molecular binding of $10^{-2}$ per second, an unbinding of $10^{-2}$ per second, and a catalytic of 1.0 per second.

For all three parametrizations, we run the following queries on the resulting traces.

**Undoing S45, T41, S37 and S33 Phosphorylation.** Considering phosphatases undoing the phosphorylation of sites, does this happen to all agents? Does it happen to just a few agents? What is the distribution of dephosphorylation events per agent? (Fig. 4)

```plaintext
match e : { c:Cat ( S45\text{p→u} ) }    match e : { c:Cat ( T41\text{p→u} ) } 
return agent_id ( c ), time [ e ]    return agent_id ( c ), time [ e ]
match e : { c:Cat ( S37\text{p→u} ) }    match e : { c:Cat ( S33\text{p→u} ) } 
return agent_id ( c ), time [ e ]    return agent_id ( c ), time [ e ]
```

**Wait Times.** What is the distribution of times spent between the first phosphorylation on an agent, and the time it gets degraded? (Fig. 5)

```plaintext
match i : { c: Cat+ } 
and first p : { c: Cat ( S45\text{u→p} ) } after i 
and first d : { c: Cat– } after p 
return time [ d ] - time [ p ] (3)
```

_About this Query._ Agent creation and destruction is expressed by suffixing agent names with + and −, respectively.
Component Size and Enzyme Identity. Where do the phosphorylation steps that actually lead to degradation occur? Do they happen mostly on large complexes? What is the composition in units of Axn and APC of the complexes where the phosphorylation events leading to degradation took place? What is the distribution of kinase identifiers for the last phosphorylation events that lead to degradation? (Fig. 6)

```
macth d : { c : Cat → } 
and last p1 : { c : Cat ( S45^1_a → p ), k1 : CK1 ( c^1 ) } before d 
and last p2 : { c : Cat ( T41^1_a → p ), k2 : GSK ( c^1 ) } before d 
and last p3 : { c : Cat ( S37^1_a → p ), k3 : GSK ( c^1 ) } before d 
and last p4 : { c : Cat ( S33^1_a → p ), k4 : GSK ( c^1 ) } before d 
return agent_id ( k1 ), count ( component [•p1] ( k1 ), "Axn", "APC" ), 
agent_id ( k2 ), count ( component [•p2] ( k2 ), "Axn", "APC" ), 
agent_id ( k3 ), count ( component [•p3] ( k3 ), "Axn", "APC" ), 
agent_id ( k4 ), count ( component [•p4] ( k4 ), "Axn", "APC" )
```

About this Query. The component state measure computes the connected component that contains an agent in a mixture. It returns a set of agents $S$. The count function takes such a set $S$ along with $n$ strings denoting agent types and returns an $n$-tuple of integers indicating how many agents of each type appear in $S$.

5.2 Results and Interpretation

Distribution of Undo Events per Agent. To study the effect of adding phosphatase, we look at the distribution of dephosphorylation events per agent in Fig. 4. S45 is the first residue to be modified in the causal chain leading to degradation; S37 is the last. Based on the 1:1 system, it is surprising to see increasing the phosphatase level five-fold maintains a similar total number of dephosphorylation events (compare curves’ integrals). However, their distribution is quite different. Interestingly, increasing the amount of kinase to 1:5 led to decrease in dephosphorylation events, even though the dephosphorylation enzyme’s abundance and rates were kept at the same levels. It is also worth noting, the 1:1 saw almost 30 thousand dephosphorylation events of S45, occurring on a shrinking pool of at most 500 copies of Cat. Clearly certain agents are caught in the undo loop; some specific agents are getting dephosphorylated almost 800 times. It is worth noting these levels of dephosphorylation imply a comparable number of phosphorylation events.

To answer the question that motivated this query, for S45 under 1:5 regime, most agents don’t get sabotaged by the phosphatase: the blue line is quite flat. Decreasing the amount of kinase changes this, and under a 1:1 regime some
agents get undone multiple times, a quarter seeing upwards of hundreds of undo events (e.g. from id 300 onward). Increasing the phosphatase to a 5:1 regime further exacerbates this, with over half the agents receiving undo events hundreds of times. The unavailability of phosphorylated S45 in turn inhibits the phosphorylation of T41, and so forth to S37 and S33. It is worth noting that, based on the 1:1 system, increasing the phosphatase five-fold decreases the number and extent of advanced dephosphorylation events, such as S33 and S37. Paradoxically, increasing the kinase five-fold has this same effect. We attribute the former to decreased availability of the intermediate phosphorylated states (i.e. if T41 is not phosphorylated, S37 can’t be phosphorylated, ergo can’t be dephosphorylated), and the latter to increased throughput to degradation (i.e. Cat is not around for long enough to get dephosphorylated, as once it gets fully phosphorylated it quickly proceeds to get degraded).

We call attention to the number of agents whose final sites got dephosphorylated (Fig. 4), vs. the number of agents who got degraded (Fig. 7, in Appendix A). The 1:5 or 1:1 systems both degraded over 450 agents each, but the former undid around 160 agents (Fig. 4 S37, domain of blue curve) while the latter undid over 350 (Fig. 4 S37, domain of red curve). For the 1:5 and 5:1 systems, both undid around 160 agents (Fig. 4 S37, domain of blue and yellow curves), but the former degraded over 450 agents (Fig. 7, blue curve) while the latter less than 50 (Fig. 7, yellow curve). This argues the notion of efficiency (e.g. minimizing the amount of undo steps) can’t readily be inferred from the throughput of the system.

![Fig. 4. Distribution of dephosphorylation events per agent. Each time an agent gets dephosphorylated, its ID is registered. After sorting, we plot the distribution of these IDs for two residues in the three parameter regimes. The area under the curve is also presented on each legend. S45 is the first residue to get phosphorylated, S37 (along with S33) is the last. (Color figure online)](image-url)
**Wait Times.** Looking at the distribution of wait times (Fig. 5), from first phosphorylation to degradation, we note the bulk of degradation events occur rapidly, in less than 50 s. Worth noting that, from the 1:1 regime, increasing the amount of kinase five-fold marginally reduced wait times.

![Distribution of wait times from first phosphorylation until degradation.](image)

**Fig. 5.** Distribution of wait times from first phosphorylation until degradation. The sum of the bins is presented in the legend of each plot, and corresponds to the total number of degradation events, matching what is seen on Fig. 7. The height of each bin represents the number of agents that waited the bin’s position (in seconds) since they were first modified until they were degraded.

**Complex Composition.** A way of looking at the question of complex contribution is to query the size of the complex at the last phosphorylation event before degradation. Taking S45 as representative of all the other residues, we plot the size of the complex, in terms of Axn and APC, at the time the final S45 occurred. Overall, we see a broad distribution of sizes, with some phosphorylation events occurring in large complexes (i.e. >80 Axn, >40 APC), but a significant number occurring in far smaller complexes (i.e. <10 Axn, <10 APC). Changing the parameter regime of kinase to phosphatase does not seem to alter this behavior significantly.

### 5.3 Summary of Findings

1. The number of undo events does not inform us of overall throughput (contrast Figs. 4 and 7).
Fig. 6. Composition of the complex, in terms of Axn and APC components, at the last event where Cat got S45 phosphorylated before being degraded. The number of points corresponds to the number of degradation events. The points of this scatter plot have been nudged with a random noise factor of 0.2 to increase visual perception of discrete points where the data overlap.

2. How a step may be affected by changing abundances depends greatly on its upstream context (Fig. 4).

3. Entities that got degraded waited a short while since their first modification (Fig. 5), and yet most modifications were futile (Fig. 4).

4. We can’t argue that giant complexes, nor small complexes, nor medium complexes, are the sole entities responsible for performing the effective (i.e. final) phosphorylation events (Fig. 6); the distribution of complexes is wide, and they all contribute to the kinetics.

The capacity of querying a simulation’s trace offers a mechanistic description of the inner workings of our system. Since this description uses the vocabulary of molecular biology, it can greatly inform the search for drug targets.

For example, in our setup, complexes with over 60 copies of Axn and over 40 copies of APC contributed a large amount of degradation events (Fig. 6). Considering there were a total of 100 copies of each scaffold, these large complexes are giant components, having recruited the majority of scaffolds into a single entity. If a single entity is contributing an amount of degradation events comparable to the rest of the mixture, it means its effective catalytic rate is greater than that of smaller entities. One could therefore reduce overall degradation of Cat.
by destabilizing any of the three scaffold interactions (i.e. Axn-Axn, APC-APC, Axn-APC), without affecting the enzymes directly. Since these enzymes are also involved in metabolism, it would be desirable to avoid affecting their behaviors outside our pathway of interest.

Beyond Wnt, our approach can serve to add an analytical dimension to the phase-transition model suggested by Pronobis et al. [12]. If the mixture transitions from an fragmented regime into an aggregated one, the aggregated regime is expected to enjoy greater local concentrations of enzymes and substrates within the large components. Through our trace query language, we can assay the size of components as they contribute catalytic events. This allows us to quantify the degradation activity inside and outside the distinct phases by querying the size of the scaffold complex at the times of enzymatic modifications. Were one to make a model of the specific setup and chimeric protein of Pronobis et al., we would expect the distribution of events to reveal the distinct phases suggested by the authors.

Our trace query language could also aid in the characterization of the aggregate-with-tentacles model of Anvarian et al. [1]. The gain-of-function mutations identified by the authors create novel binding capacities. By distinguishing bi-molecular from uni-molecular interactions, one can model local concentration effects. Specifically, one can model the effect of having novel binding sites on the complex’ tentacles, in close proximity by nature of being loaded on the same complex. One can therefore model the growth of the complex as additional binding sites become active (e.g. by setting a binding rate to greater than zero). Were one to make a model informed by the specific experimental setup used by the authors, one could quantify the change in unbound state as additional binding sites become active.

6 Related Works

Languages already exist to collect, filter and aggregate data from simulation. For example, the Kappa simulator [2] features an intervention language that allows taking repeated conditional measurements during simulation, possibly updating a global state every time. Chromar [11] proposes an extended language with similar capabilities, where queries can be defined in a more functional style from a selector and a fold operation. Finally, query languages have been proposed [10,13] that are inspired by the structured query language (SQL).

All these languages are similar in the sense that (i) queries can only evaluate expressions in the scope of a single state or transition and (ii) only population-level quantities can be measured. In contrast, queries in our proposed language can (i) match trace patterns that consist in multiple transitions acting on common agents and (ii) measure and compare the state of individual agents, possibly at different points in time. As a consequence, most of the example queries shown in this paper could not be expressed using preexisting query languages.

Thanks to its ability to match complex trace patterns, our query language may be especially interesting to use in combination with causal analysis [4,6].
Indeed, the pathways uncovered by causal analysis can be regarded as trace patterns and then matched using our query language. This may be useful to measure the relative frequency of pathways in different settings, but also to analyze how individual agents participating to a pathway evolve as this pathway unfolds. For example, query (4) in Sect. 5 compares the composition of the complex at which $\beta$-catenin is attached at different points of the pathway leading to its degradation.

7 Conclusion

How can one explore a question like “which complexes contribute to kinetics”? Experimental biologists have been using labeling techniques for decades, but implementing this in a modeling framework requires being able to track individual agents, and query particular events. Implementing a framework to query events on the trace of an agent and rule simulation seems a natural way of tackling these classes of problems.

Moreover, once a sufficiently rich mechanistic model is available, questions on mechanism arise. For a subset of these, a satisfying answer will require a change of vocabulary; the explanations desired use the individual’s lexicon (e.g. it bound, it unbound, it got dephosphorylated 800 times), rather than a whole system lexicon (e.g. the abundance changed from 500 to 50). Thus, rather than tracking the whole model’s behavior (akin to a top-down approach), one needs to focus on agents, and observe their individual experiences (akin to a bottom-up approach). These approaches are complementary, as they explore a model’s intricacies from very different viewpoints. We hope that the query language proposed in this paper will contribute to make agent-centric analysis more widespread and accessible.

Acknowledgments. This work was sponsored by the Defense Advanced Research Projects Agency (DARPA) and the U.S. Army Research Office under grant numbers W911NF-14-1-0367 and W911NF-17-1-0073.

A Use Case Appendix

Concentration Time Traces. From the output of the simulator, we get the evolution of the abundance of Cat through time. In Fig. 7, we can see that the systems with low phosphatase behave similarly, even though one has five times the amount of kinases than the other (blue vs red traces). In contrast, the system with high phosphatase shows markedly less degradation of Cat; where the other two systems degraded around 450 units, this one has only degraded 23. From this whole-system view, it would seem the amount of phosphatase is more critical than the amount of kinase: based on the 1:1 system, increasing the kinase five-fold has little effect, whereas increasing the phosphatase has a more dramatic effect.
Fig. 7. Tracking the abundance of agent cat through the simulation. At time $T = 0$, the agents are introduced, all in monomeric form. The simulation was stopped after five hundred simulated seconds. In this legend and throughout the figures, “ph” stands for phosphatase, “ki” stands for kinase, and numbers indicate agent multiplicity. Thus “10 ph: 50 ki” means the system with 10 units of phosphatase and 50 of kinase. (Color figure online)

References


In this chapter, I present briefly two tools I wrote for analyzing and visualizing Kappa entities (section 3.1), and presenting Kappa code in a structured (i.e., highlighted) fashion (section 3.2).
3.1 Kappa Snapshot Analyzer

3.1.1 Introduction

The Kappa simulator allows one to output the state of the mixture at any given time. This allows one to measure the changes of entities in the mixture for which an observable variable can not be defined, such as pleomorphic complexes. However, the analysis of such mixture states is far from trivial. I have developed a series of Python classes and scripts to tackle this problem—a suite I call KaSaAn. It is open source and available at https://github.com/hmedina/KaSaAn.

3.1.2 Core classes

The tools rely on a hierarchy of classes. A KappaSnapshot contains metadata (like snapshot time, run identifier, event number), a possibly empty collection of instances of KappaToken, and a collection of instances of KappaComplex at various abundances. A KappaComplex is composed of multiple instances of KappaAgent. A KappaAgent contains metadata like its name, and a set of sites: instances of either KappaPort or KappaCounter. A KappaPort has metadata like its name, and possibly internal state data (e.g. phosphorylated) and/or bond state data (e.g. using bond #7). In contrast, a KappaCounter has metadata like its name, and an integer value. For an in-depth exposition of the methods exposed by these classes, see the associated readme file.

The object-oriented nature of this design yields a syntax that is hierarchical. For example, querying the amount of $\beta$-catenin with S37 in phosphorylated state and unbound at the armadillo repeats, for the (first) largest complex in the mix becomes:
from KaSaAn.core import KappaSnapshot, KappaAgent

my_agent = KappaAgent('bCat(S37{ph}, ARM[.]')
my_snap = KappaSnapshot('snapshot.ka')
my_largest = my_snap.get_largest_complexes()
my_largest[0].get_number_of_embeddings_of_agent(my_agent)

With this design, inclusion testing is straightforward. For example, if one wanted to know the amount of APC in every complex where the above β-catenin appears, one could do as follows:

my_list = []
other_agent = KappaAgent('APC()')
for complex in my_snap.get_all_complexes():
    if my_agent in complex:
        my_list.append(complex.get_number_of_embeddings_of_agent(other_agent))

3.1.3 Scripts and functions

Using the classes outline above, I wrote a suite of scripts for automating various analysis (see KaSaAn/functions). For convenience, I also package some of these as executable scripts, available from the command line (see KaSaAn/scripts) using Python’s SetupTools entry-points. These are all self-documenting tools, and so invoking them with the -h or --help flags will print out their usage, parameters, and description.

Future work Depending on time commitments, I may devote work to the visualization code for rules. My ambition is to have the same codebase visualize rules, contact maps, as well as snapshots, with fidelity to the individual sites and states. I have a proof-of-principle implementation that shows promise for a subset of the Kappa language, which I may revisit as, or if, needed.
3.2 **Kappa Code Highlighter**

I wanted to present the code for the Wnt model in a readable fashion. Specifically I wanted to obviate to the reader the operations being performed, as well as the agents involved. I therefore rewrote the model using the Kappa edit notation, which explicitly marks what part of the pattern is being modified. My rules frequently contain a large number of causal constraints, often in the form of sites in specific states (e.g. phosphorylated). This made the Kappa lines long. To visually identify Kappa operations, I needed a code highlighter, specifically one that could produce or be compatible with \TeX{} code.

There are two \LaTeX{} packages that are frequently used for code highlighting, Listings and Minted. The former is simpler, but didn’t support some features I required, notably concerning the lexing engine amenability to work with a language like Kappa. The latter uses the Pygments Python package, and its lexing engine can use a parsers similar to the regex-based one already used in the Kappa GUI. Moreover, Pygments is used by many websites and services, from Overleaf to GitHub and even Wikipedia.

### 3.2.1 Writing a decent lexer

Adding languages is documented in the Pygments documentation. The first challenge was constructing a lexer that will read the input code and generate a list of tokens (i.e. types) and locations – a token specifying a text style for the given code location. Pygments uses lexers written as regular expressions. Kappa is not a regular language, and so a simple regex lexer would not be able to parse it.
correctly; indeed, the simple parser used for the GUI has issues distinguishing some nuances of the Kappa syntax. For complex languages, Pygments supports multi-state lexers, and my parser utilizes 6 regex states to correctly parse the language.

### 3.2.2 Having the lexer do something

The second challenge lies in Kappa not being a programming language like Python or C#: the built-in tokens used by Pygments were ill-fitting. I took advantage of the tree-like structure of Kappa patterns and the token inheritance scheme to build a hierarchy of new tokens matching components of a Kappa expression. This tree traversal resulted, expectedly, in an explosion of leaves, yielding a very large number of tokens, over 60.

### 3.2.3 Making the something be useful

Now with a lexer that issues a list of appropriate tokens for every component of a Kappa expression, the final task was to make a style-sheet. This is effectively a Python dictionary that defines a style for every node in the token tree, with leaves inheriting styles from their parents. My preferred style bolds agent names in rules (but not in bond typing), color-codes edit notation operations, and prints in red the commas separating agents (but not those inside agent signatures):

```plaintext
// enzymatic reaction with atomic operations
'bind' enzyme(site[1..1]), substrate(site[1..1]) @ 'binding rate'
'modify' substrate(site[site.enzyme], state{un/ph}) @ turnover_rate
'release' enzyme(site[1..1]), substrate(site[1..1], state{ph}) @ 'unbind rate'
// zero-th order production, first order degradation of substrate
```
While this architecture allows a very powerful and customizable coloring of Kappa expressions, it was too much idiosyncrasy for the main Pygments distribution, and pull request #1293 was rejected. So I repackaged my work as a plugin, available at hmedinaPygments_Kappa_plugin. Installation, usage, classes, and extensions are all documented in that repository.
Combinatorial protein-protein interactions
on a polymerizing scaffold

The full Wnt model is combinatorially complex. I devised simpler models-of-a-model that would be easier to analyze. This article is our treatment of the first. It is inspired by the core
enzymatic interactions in the β-catenin “destruction complex”.

When building the Wnt model, I wanted to be faithful to the biology of the proteins involved. Therefore I zealously sought out binding sites. It was then I realized that the enzymes that phosphorylate β-catenin, i.e. CK1ε, GSK3β, do not actually bind it. It is Axin who binds β-catenin, and brings it into proximity of those enzymes. Moreover, Axin polymerizes in a head-to-tail manner using its DIX domain. Considering the small size of the binding sites, the large disordered regions in Axin, and the way the DIX domain polymerizes, all suggest the Axin polymer could act as a polymeric scaffold, capable of recruiting some amount of enzymes and substrates into a pseudo-compartment, increasing their local concentration. This would be akin to a programmable surface.

Andrés Ortíz (CalTech) developed a partition-function approach to calculating the equilibrium concentration of complexes, specifically homo-polymers. By its nature, this approach lent itself to the analysis of systems where an enzyme utilizes a polymerizing scaffold to recruit its substrate –a simple but surprisingly rich toy model. The article is available on the q-bio arXiv as 1907.11533 and at the Proceedings of the National Academy of Sciences website under DOI 10.1073/pnas.1912745117.

Although the core interactions belong the canonical Wnt signaling cascade, the β-catenin “destruction complex” is involved in the regulation of RAS, NFκB, and even P53. Understanding the kinetics of this complex is likely to advance our understanding of other signaling cascades, at the very least through their cross-talk. Moreover, other signaling cascades utilize polymerizing scaffolds and disordered proteins. A faithful reconstruction of them might reveal this same mechanism at play outside canonical Wnt signaling.
Combinatorial protein–protein interactions on a polymerizing scaffold

Andrés Ortiz-Muñoz\textsuperscript{a,1}, Héctor F. Medina-Abarca\textsuperscript{b,1}, and Walter Fontana\textsuperscript{b,2,3}

\textsuperscript{a}Division of Biology and Bioengineering, California Institute of Technology, Pasadena, CA 91125; and \textsuperscript{b}Department of Systems Biology, Harvard Medical School, Boston, MA 02115

Edited by Peter Schuster, University of Vienna, Vienna, Austria, and approved December 24, 2019 (received for review July 24, 2019)

Scaffold proteins organize cellular processes by bringing signaling molecules into interaction, sometimes by forming large signalosomes. Several of these scaffolds are known to polymerize. Their assemblies should therefore not be understood as stoichiometric aggregates, but as combinatorial ensembles. We analyze the combinatorial interaction of ligands loaded on polymeric scaffolds, in both a continuum and discrete setting, and compare it with multivalent scaffolds with fixed number of binding sites. The quantity of interest is the abundance of ligand interaction possibilities—the catalytic potential $Q$—in a configurational mixture. Upon increasing scaffold abundance, polymerizing systems are known to first increase opportunities for ligand interaction and then to show a decrease in $Q$ at high densities of the ligand scaffold. We adapt this presentation to a combinatorial setting by viewing each different scaffold as a distinct scaffold. The polymerizing system stands out in that the dependency of $Q$ on protomer concentration switches from being dominated by a first order to a second order term within a range determined by the polymerization affinity. This behavior boosts $Q$ beyond that of any multivalent scaffold system. In addition, the subsequent drop-off is considerably mitigated in that $Q$ decreases with half the power in protomer concentration than for any multivalent scaffold. We explain this behavior in terms of how the concentration profile of the polymer-length distribution adjusts to changes in protomer concentration and affinity. The discrete case turns out to be similar, but the behavior can be exaggerated at small protomer numbers because of a maximal polymer size, analogous to finite-size effects in bond percolation on a lattice.

Protein–protein interactions underlying cellular signaling systems are mediated by a variety of structural elements, such as docking regions, modular recognition domains, and scaffold or adapter proteins (1, 2). These devices facilitate the evolution and control of connectivity within and among pathways. In particular, the scaffolding function of a protein can be conditional upon activation and serve to recruit further scaffolds, thus creating opportunities for network plasticity in real time. Scaffolds are involved in the formation of signalosomes, which are transient protein complexes that process and propagate signals. A case in point is the so-called “destruction complex” that tags $\beta$-catenin for degradation in the canonical Wnt pathway. $\beta$-Catenin is modified by CK1$\alpha$ and GSK3$\beta$ without binding any of these kinases directly but interacting with them through an Axin scaffold (3, 4). In addition, the DIX domain in Axin allows for oriented Axin polymers (5, 6), while APC (another scaffold) can bind multiple copies of Axin (7), yielding Axin–APC aggregates to which kinases and their substrates bind. By virtue of their polymeric nature, such scaffold assemblies have no defined stoichiometry and may only exist as a heterogeneous combinatorial ensemble (8, 9)—also called “pleomorphic ensemble” (10)—rather than a single well-defined complex. Deletion of Axin’s DIX domain abolishes degradation of $\beta$-catenin (5), and mutations in APC that drive familial adenomatous polyposis map to truncations reducing the number of SAMP repeats at which APC binds Axin (11). These observations suggest a possible link between the size distribution of scaffolding aggregates and disease.

Interest in intracellular phase separation phenomena has increased since the discovery of P body dissolution/condensation in Caenorhabditis elegans (12). Much attention has been given to the physics underlying sol–gel transitions and polymerization (13–15). Here, we focus on the combinatorial aspects of ligand interactions on a (noncovalently) polymerizing scaffold, in particular, since scaffold-mediated interactions are subject to the prozone or “hook” effect (16–18): low scaffold concentrations promote interactions between ligands, but high concentrations oppose them by isolating ligands on different scaffold molecules. The main objective of our analysis is to gain insight into how this effect plays out in the context of a polymerizing-scaffold system and whether it affords opportunities for regulation. We proceed by defining and analyzing a simple model at a level of abstraction that only encapsulates combinatorial features without explicit taking into account spatial constraints arising from polymer conformation.

The Polymerizing-Scaffold System

Let $S$ (the scaffold) be an agent with four distinct binding sites $\{a,b,x,y\}$. At site $y$, agent $S$ can reversibly bind site $x$ of another $S$ with affinity $\sigma$, forming (oriented) chains. For the time being, we exclude the formation of rings. Sites $a$ and $b$ can reversibly bind an agent of type $A$ (the enzyme) and of type $B$ (the substrate) with affinities $\alpha$ and $\beta$, respectively. All binding interactions are independent. When the system is closed, the total concentrations

Significance

Scaffold proteins play an important role in the control and evolution of cellular signaling processes by brokering interactions among docking proteins. In recent years it has become apparent that many scaffolds can form oligomers. We explore theoretically the combinatorial impact of scaffold polymerization on promoting interactions. It is well known that increasing scaffold abundance first expands opportunities for ligand interaction but then collapses them as ligands become separated on different scaffold instances. The polymerizing system stands out by significantly heightening interaction possibilities in the promoting phase and considerably mitigating their collapse. While the qualitative behavior of the system is robust, parameter changes—achievable through the very processes the system mediates—can considerably sculpt its quantitative behavior.

Author contributions: A.O.-M., H.F.M.-A., and W.F. designed research, performed research, contributed new reagents/analytic tools, and wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission.

Published under the PNAS license.

1 A.O.-M. and H.F.M.-A. contributed equally to this work.
2 To whom correspondence may be addressed. Email: walter.fontana@hms.harvard.edu.
3 Present address: Informatique et sciences numeriques, Collège de France, 75231 Paris Cedex 05, France.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.1912745117/-/DCSupplemental.


2930–2937 | PNAS | February 11, 2020 | vol. 117 | no. 6

www.pnas.org/cgi/doi/10.1073/pnas.1912745117
of $A$, $B$, and $S$ are given by $t_A$, $t_B$, and $t_S$. This setup allows for a variety of configurations, such as shown in Fig. 1A. We posit that each enzyme $A$ can act on each substrate $B$ bound to the same complex. We refer to the number $p q$ of potential interactions enabled by a configuration with sum formula $A_p S_q B_r$ as that configuration’s “catalytic potential” $Q$. By extension, we will speak of the catalytic potential $Q$ of a mixture of configurations as the sum of their catalytic potentials weighted by their concentrations.

If we assume that the assembly system equilibrates rapidly, the rate of product formation is given by $Q k_{cat}$, with $k_{cat}$ the catalytic rate constant and $Q$ the equilibrium abundance of potential interactions between $A$ and $B$ agents. Rapid equilibration is a less realistic assumption than a quasi-steady state but should nonetheless convey the essential behavior of the system. We first provide a continuum description of equilibrium $Q$ in terms of concentrations (which do not impose a maximum polymer length) and then a discrete statistical physics treatment for the average equilibrium $Q$ (where $t_S$ is a natural number and imposes a maximum polymer length).

In the present context, molecular species $Y_i$ assemble from $T = 3$ distinct building blocks (“atoms”) $X_i$ through reversible binding interactions. The $Y_i$ have a graphical (as opposed to geometric) structure (Fig. 1). We denote the number of species $Y_i$ by $\mu_i$. The equilibrium concentration $y_i$ of any species $Y_i$ can be obtained by recursion over the assembly reactions as $y_i = \xi_i \prod_{j=1}^i y_j^{a_{ij}}$, where $\xi_i = 1/\omega_i$ $\prod_{j \in P} K_{ij}$ is the exponential of the free-energy content of $Y_i$, where $\omega_i$ denotes the number of symmetries of $Y_i$, which, in our case, is always 1 because the polymers are oriented. The product runs over a sequence of reactions $r$ that form an assembly path $P$ of $Y_i$. In equilibrium, it is irrelevant which $P$ one chooses. $K_{ij} \in \{\alpha, \beta, \sigma\}$ is the equilibrium constant of the $r$th reaction, and the $\xi_i$ are the equilibrium concentrations of free atoms of type $j$. Aside from the symmetry correction, $\xi_i$ is the exponential of a sum of binding energies, for example, $\xi_i = \alpha^\beta^\sigma$ for a $Y_i$ that contains $p$ bonds between $A$ and $S$, $q$ bonds between $B$ and $S$, and $r$ bonds between $S$ protomers.

Consider first the polymerization subsystem. From what we just laid out, the equilibrium concentration of a polymer of length $l$ is $\sigma^{l-1} s^l$, where $s$ is the equilibrium concentration of monomers of $S$. Summing over all polymer concentrations yields the total abundance of entities in the system,

$$W(s) = \sum_{l=1}^{\infty} \sigma^{l-1} s^l = s/(1 - \sigma s).$$

Eq. 1 yields the conservation relation, $t_S = s dW(s)/ds$, from which we obtain $s$ as:

$$s = \frac{1}{4\sigma} \left(\sqrt{4 + 1/(\sigma t_S)} - \sqrt{1/(\sigma t_S)}\right)^2.\quad[2]$$

Using Eq. 2 in $s^{-1} s^l$ yields the dependence of the polymer-size distribution on parameters $t_S$ and $\sigma$. $W(s)$ has a critical point at $s_{cr} = 1/\sigma$, at which the concentrations of all length classes become identical. It is clear from Eq. 2 that $s$ can never attain that critical value for finite $\sigma$ and $t_S$.

The Chemostatted Case

In a chemostatted system, $s$ can be fixed at any desired value, including the critical point $1/\sigma$. At this point, ever more protomers are drawn from the $S$ reservoir into the system to compensate for their incorporation into polymers. We next consider the system with ligands $A$ and $B$ held fixed at concentrations $a$ and $b$. Let $A_p S_q B_r$ be the sum formula of a scaffold polymer of length $n$ with $p A$ agents and $q B$ agents. There are $(\binom{n}{3})^{(n)} p q$ such configurations, each with the same catalytic potential $Q = p q$. Summing up the equilibrium abundances of all configurations yields

$$W(s, a, b) = a + b + \frac{s(1 + \alpha a)(1 + \beta b)}{1 - \sigma s(1 + \alpha a)(1 + \beta b)}.\quad[3]$$

Eq. 3 corresponds to the $W(s)$ of ligand-free polymerization, Eq. 1, by a coarse-graining that erases the ligand-binding state of scaffolds, i.e., by dropping terms not containing $s$ and substituting $s$ for $(1 + \alpha a)(1 + \beta b)$. Eq. 3 indicates that, at constant chemical potential for $A$, $B$, and $S$, the presence of ligands lowers the critical point of polymerization to $s_{cr} = 1/(\sigma(1 + \alpha a)(1 + \beta b))$ because, in addition to polymerization, free $S$ is also removed through binding with $A$ and $B$.

$Q_{poly}$, the $Q$ of the system, is obtained by summing up the $Q$ of each configuration weighted by its equilibrium concentration (SI Appendix, section 1). Using $W$, we compute $Q_{poly}$ as

$$Q_{poly} = ab \frac{\partial^2}{\partial a \partial b} W = \alpha \alpha a b s \frac{1 + s(1 + \alpha a)(1 + \beta b)}{(1 - \sigma s(1 + \alpha a)(1 + \beta b))^2}.\quad[4]$$

Note that $Q_{poly}$ inherits the critical point of $W$. The behavior of the chemostatted continuum model is summarized in Fig. 2. $Q_{poly}$ (red) diverges as the polymerization system approaches the critical point. Fig. 2A, Inset shows the scaffold-length distribution at the black dot on the $Q_{poly}$ profile. The red dotted curve reports the length distribution in the presence

![Fig. 1. Scaffold types. (A) Protopmers $S$, each binding an enzyme $A$ and a substrate $B$ with affinities $\alpha$ and $\beta$, respectively, polymerize with affinity $\sigma$ to yield a distribution of complexes. (B) A monovalent scaffold does not polymerize and has only one binding site for $A$ and $B$ each. (C) An $n$-valent (or multivalent) scaffold is a nonpolymerizing scaffold with $n$ binding sites for $A$ and $B$ each. Here, $n = 3$. The catalytic potential $Q$ of a configuration is the number of possible interactions between $A$ and $B$ agents bound to the same complex: 12 in $A$, 1 in $B$, and 4 in $C$.](https://www.pnas.org/content/117/6/2931)

Ortiz-Muñoz et al.  
PNAS | February 11, 2020 | vol. 117 | no. 6 | 2931
of ligands, \( [A \cdot S_b \cdot B_s] = \sigma^{-1}(\sigma s(1 + \alpha a)(1 + \beta b))^k \), whereas the black dotted curve reports the length distribution in the absence of ligands, \( s_s = [S_s] = \sigma^{k-1}s^k \). The presence of \( A \) and \( B \) shifts the distribution to longer chains. The blue curve in Fig. 2A shows the catalytic potential of the monovalent scaffold, \( \sigma = 0 \). It increases linearly with \( s \) but at an insignificant slope compared with the polymerizing case, which responds by raising the size distribution, thus drawing in more \( S \) from the reservoir to maintain a fixed \( s \); this, in turn, draws more \( A \) and \( B \) into the system. In Fig. 2B, \( s \) is fixed and \( b \), the substrate concentration, is increased. The green straight line is the Michaelis–Menten case, which consists in the direct formation of an \( AB \) complex and whose \( Q = \alpha a b \) is linear in \( b \). The red line is the polymerizing-scaffold system whose \( \kappa_s \) can be attained by just increasing \( b \) (Eq. 4). All else being equal, there is a \( b \) at which more substrate can be processed than through direct interaction with an enzyme. The slope of the monovalent scaffold (blue) is not noticeable on this scale.

**The Continuum Case in Equilibrium**

We turn to the system with fixed resources \( t_S \), \( t_A \), and \( t_B \), expressed as real-valued concentrations. Eq. 4 for \( Q_{\text{poly}} \) is now evaluated at the equilibrium concentrations \( s \), \( a \), and \( b \) of the free atoms. These are obtained by solving the system of conservation equations, \( t_S = s \frac{\partial W}{\partial s} \), \( t_A = a \frac{\partial W}{\partial a} \), and \( t_B = b \frac{\partial W}{\partial b} \) (solutions in SI Appendix, section 1). The orange curve in Fig. 3A depicts the saturation curve of the catalytic potential \( Q_{\text{direct}} \) of the Michaelis–Menten mechanism for a fixed concentration \( t_A \).
of enzyme as a function of substrate \( t_B \). The green curves are saturation profiles of the polymerizing-scaffold system at varying protomer abundances \( t_s \) under the same condition. As in the chemostated case, beyond some value of \( t_s \), the catalytic potential of the polymerizing system exceeds that from direct interaction.

\( Q_{\text{poly}} \) can be modulated not only by the protomer concentration \( t_s \) but also the protomer affinity \( \sigma \) (Fig. 3B). Increasing \( t_s \) improves \( Q_{\text{poly}} \) dramatically at all affinities up to a maximum after which enzyme and substrate become progressively separated due to the prozone effect. At all protomer concentrations, in particular, around the maximizing one, \( Q_{\text{poly}} \) always increases with increasing affinity \( \sigma \).

**Comparison with Multivalent Scaffold Systems.** With regard to \( Q \), a polymer chain of length \( n \) is equivalent to a multivalent scaffold agent \( S(n) \) with \( n \) binding sites for \( A \) and \( B \) each. It is therefore illuminating to compare the polymerizing system with multivalent scaffolds and their mixtures.

The equilibrium concentration of configurations \( A_p S(n) B_q \) for an \( n \)-valent scaffold can be calculated by exploiting the independence of binding interactions (SI Appendix, section 2).

The calculation yields as a general result that the catalytic potential for an arbitrary scaffolding system consists of two factors:

\[
Q = p(t_{\text{t1}}, t_A, \alpha)p(t_{\text{t2}}, t_B, \beta) Q_{\text{max}}(t_s). 
\]

This dimensionless function \( p(t_{\text{t1}}, t_X, \gamma) \) denotes the equilibrium fraction of \( X \)-binding sites, with total concentration \( t_X \), that are occupied by ligands of type \( X \), with total concentration \( t_X \), interacting with affinity \( \gamma \):

\[
p(t_{\text{t1}}, t_X, \gamma) = \frac{\gamma t_X - \gamma t_{\text{t1}} - 1 + \sqrt{4\gamma t_X + (\gamma t_X - \gamma t_{\text{t1}} - 1)^2}}{\gamma t_X + \gamma t_{\text{t1}} + 1 + \sqrt{4\gamma t_X + (\gamma t_X - \gamma t_{\text{t1}} - 1)^2}}.
\]

This expression is the well-known dimerization equilibrium, computed at the level of sites rather than scaffolds and taken relative to \( t_{\text{t1}} \) (SI Appendix, section 2).

Factor \( I \) depends on the total concentration of ligand-binding sites (for each type) but not on how these sites are partitioned across the agents providing them. For example, a multivalent scaffold \( S(n) \) present at concentration \( t_{\text{t2}} \), provides \( t_{\text{t2}} = n t_{\text{t2}} \) binding sites and the probability that a site of any particular agent is occupied is the same as the probability that a site in a pool of \( n t_{\text{t2}} \) sites is occupied. For a heterogeneous mixture of multivalent scaffold agents, we have \( t_{\text{t1}} = \sum_{i=1}^{n} t_{\text{t1}(i)} \) for a polymerizing system in which each protomer \( S \) exposes one binding site, we have \( t_{\text{t1}} = t_s \).

Factor \( II \) is the maximal \( Q \) attainable in a scaffolding system. This factor depends on how sites are partitioned across scaffold agents with concentrations \( t_s = (t_{s(1)}, \ldots, t_{s(n)}) \) but does not depend on ligand-binding equilibria. For example, a system of multivalent agents at concentrations \( t_s \) has \( Q_{\text{max}} = \sum_{i=1}^{n} \frac{1}{s-1} t_{s(i)} \). The polymerizing-scaffold system is analogous, but \( n = \infty \) and the \( t_{s(i)} \) are determined endogenously by aggregation: \( t_{s(i)} = n_s = \sigma^{-1} s' \). This yields simple expressions for the catalytic potential of a polymerizing scaffold, \( Q_{\text{poly}} \), and multivalent scaffold, \( Q_{\text{multi}} \):

\[
Q_{\text{poly}} = p(t_s, t_A, \alpha)p(t_s, t_B, \beta) \frac{s(1 + s)}{(1 - s)^3} \tag{6}
\]

\[
Q_{\text{multi}} = p(n t_{s(1)}, t_A, \alpha) p(n t_{s(2)}, t_B, \beta) n^2 t_{\text{t1}} \tag{5}
\]

with \( s \) in Eq. 6 given by Eq. 2. Eq. 6 is equivalent to Eq. 4. While Eq. 4 requires solving a system of mass conservation equations to obtain \( a, b, \) and \( s \), \( Q_{\text{poly}} \), as given by Eq. 6, does not refer to \( a \) and \( b \) but only to \( s \), as determined by the ligand-free polymerization subsystem. The \( Q \) that shapes the Michaelis–Menten rate law under the assumption of rapid equilibration of enzyme–substrate binding has the same structure as Eq. 5.

Comparison with Multivalent Scaffold Systems. For \( Q_{\text{multi}} \) various valencies (blue) at the same site concentration \( t_{\text{t1}} = t_s \). A log–log scale, scaffolds of arbitrary valency \( n \) exhibit a \( Q_{\text{multi}} \) whose slope as a function of \( t_{\text{t1}} \) is 1, with offset proportional to \( n \), until close to the peak. For the polymerizing scaffold, the first-order term of the series expansion of \( Q_{\text{poly}} \) is independent of the affinity \( \sigma \) (SI Appendix, section 5), whereas the second-order term is linear in \( \sigma \). Hence, for small \( t_{\text{t1}} \), the polymerizing system behaves like a monovalent scaffold, and any multivalent scaffold offers a better catalytic potential. However, as \( t_s \) increases, the equilibrium shifts markedly toward polymerization, resulting in a slope of 2, which is steeper than that of any multivalent scaffold. The steepening of \( Q_{\text{poly}} \) is a consequence of longer chains siphoning off ligands from shorter \( Q_{\text{multi}} \) (SI Appendix, section 4). All \( n \)-valent scaffolds reach their maximal \( Q_{\text{multi}} \) at the same abundance of sites \( t_{\text{t1}} = n t_{s(1)} = t_s \) and before \( Q_{\text{poly}} \). The superlinear growth in \( Q_{\text{poly}} \) of the polymerizing system softens the decline of \( Q_{\text{poly}} \) to an order \( t_{\text{t1}}^{-1/2} \) for large \( t_s \). In contrast, the decline of \( Q_{\text{multi}} \) is of order \( t_{\text{t1}}^{-1} \). In sum, the polymerizing-scaffold system catches up with any multivalent scaffold, reaches peak-Q later, and declines much slower.

The mitigation of the prozone effect begs for a mechanistic explanation, in particular, since a prozone could occur not only within each length class but also between classes. To assess the within-class prozone, we think of a length class \( k \) as if it were an isolated \( k \)-valent scaffold population at concentration \( t_{s(k)} = t_{\text{t1}} = \sigma^{-1} s' k^2 s \). Assuming equal affinity \( \alpha \) for both ligands \( A \) and \( B \), \( Q_{\text{multi}} \) at \( t_{s(k)} = k^{-1} (\alpha + (t_A + t_B)/2) \). However, when established through a polymerization system, \( t_{s(k)} = \sigma_k \leq 1/\sigma \) for any \( k \) and any \( t_{\text{t1}} \) (SI Appendix, section 2 and Fig. S1A). This means that for \( k \) up to \( \sigma / \alpha + (t_A + t_B)/2 \), the concentration \( s_k \) of polymers of length \( k \) can never exceed the concentration required for the prozone peak \( t_{s(1)} \). For the parameters used in the red curve of Fig. 4B, this value of \( k \) is about 35. To put this in perspective, in Fig. 4B at the yellow marker and at peak-\( Q_{\text{poly}} \), 98 and 68%, respectively, of all sites are organized in length classes below 10. Thus, the most populated lengths avoid the within-class prozone entirely (for example, \( k = 3 \) in Fig. 4C, green solid line). However, the actual behavior of the \( k \)th length class occurs in the context of all other classes, i.e., at site concentration \( t_s \), not just \( t_s \). In that frame, the class does exhibit a prozone (Fig. 4C, red solid line). Hence, the overall prozone of the polymerizing-scaffold system is mainly due to the distribution and ensuing isolation of ligands across length classes, not within. This “heterogeneity prozone”
becomes noticeable only when including all length classes up to relatively high $k$ because the majority of sites are concentrated at low $k$, where they are even jointly insufficient to cause a prozone (Fig. 4D).

At constant $t_S$ and in the limit $\sigma \to \infty$, $s_k$ tends toward zero for all $k$ (SI Appendix, Fig. S3C). In the $s$ dimension, unlike in the $t$ dimension, the class $s_1$ itself has a peak. As $\sigma$ increases, the $k$ of the class that peaks at a given $\sigma$ increases. Consequently, the $Q_{\text{mult}}$ of each length class in isolation will show a “fake” prozone with increasing $\sigma$, due entirely to the polymerization wave passing through class $k$ as it moves toward higher $k$ while flattening (Fig. 4C, dotted lines). Since there is no site inflation, the overall $Q_{\text{poly}}$ increases monotonically.

Effects of ligand imbalance and unequal ligand-binding affinities are discussed in the SI Appendix, section 11.

**Interaction Horizon.** The assumption that every $A$ can interact with every $B$ on the same scaffold can be relaxed by introducing an “interaction horizon,” $q_{\text{max}}(l, h)$, defined as the number $h$ of scaffold bonds within which a bound $A$ can interact with a bound $B$ on a polymer of size $l$. Thus, an $A$ can interact with at most $2h+1$ substrate agents $B$: its “left,” $h$ to its “right,” and the one bound to the same protomer. The interaction horizon only affects the $Q_{\text{max}}$ of a polymer of length $l$, thus replacing the interaction factor $l^2$ with (SI Appendix, section 6):

$$q_{\text{max}}(l, h) = \begin{cases} l(2h + 1) - h(h + 1), & \text{for } 0 \leq h \leq l - 1, \\ l^2, & \text{for } h \geq l \end{cases}.$$

In the most restrictive scenario, we assume a fixed horizon $h$, independent of $l$. With this assumption, Eq. 6 becomes (SI Appendix, section 6)

$$Q_{\text{poly}} = p(t_S, t_A, \alpha) p(t_S, t_B, \beta) \frac{\alpha + \sigma s - 2(\sigma s)^h}{(1 - \sigma s)^3}.$$  

In Eq. 7, the numerator of the $Q_{\text{max}}$ term of Eq. 6 is corrected by $-2s(\sigma s)^{h+1}$. Since $\sigma s < 1$ for all finite $t_S$ and $\sigma$, even
moderate values of $h$ yield only a small correction to the base case of a limitless horizon.

**The Discrete Case in Equilibrium**

In the discrete case, we replace concentrations with particle numbers $t_S, t_A, t_B \in \mathbb{N}$ in a specified reaction volume $V$. In this setting, we must convert deterministic equilibrium constants, such as $\sigma$, to corresponding “stochastic” equilibrium constants $\sigma_s$, through $\sigma_s = \sigma/(AV)$, where $A$ is Avogadro constant. For simplicity, we overload notation and use $\sigma$ for $\sigma_s$.

The basic quantity we need to calculate is the average catalytic potential $\langle Q_{\text{poly}} \rangle = \sum_{l, i, j} \langle \rho_{ij} \rangle$, where $\langle \rho_{ij} \rangle$ is the average number of occurrences of a polymer of length $l$ with $i$ and $j$ ligands of type $A$ and $B$, respectively. Conceptually, $\langle \rho_{ij} \rangle$ counts the occurrences of an assembly configuration $A_S B_i$ in every possible state of the system weighted by that state’s Boltzmann probability. In *SI Appendix, section 7*, we show that $\langle \rho_{ij} \rangle$ is given by the number of ways of building one copy of $A_S B_i$ from given resources $(t_S, t_A, t_B)$ times the ratio of two partition functions—one based on a set of resources reduced by the amounts needed to build configuration $A_S B_i$, the other based on the original resources. The posited independence of all binding processes in our model implies that the partition function is the product of the partition functions of polymerization and dimerization, which are straightforward to calculate (*SI Appendix, section 8*).

While exact, the expressions we derive for $\langle Q_{\text{poly}} \rangle$ (*SI Appendix, section 8, Eq. 66*) and $\langle Q_{\text{multi}} \rangle$ (*SI Appendix, section 8, Eq. 69*) are sums of combinatorial terms and therefore not particularly revealing. For numerical evaluation of these expressions, we change the size of the system by a factor $\xi$ (typically $\xi = 0.01$), i.e., we multiply volume and particle numbers with $\xi$ and affinities with $1/\xi$. Such resizing preserves the average behavior. Our numerical examples therefore typically deal with $10$ to $1,000$ particles and stochastic affinities on the order of $10^{-2}$ to $10^{-1}$ molecules.

The key aspect of the discrete case is the existence of a largest polymer consisting of all $t_S$ protomers. We refer to it as the “maximer”: no maximer exists in the continuum case because of the infinite fungibility of concentrations (Fig. S9). Since there is only one maximer for a given $t_S$, its expectation is the probability of observing it: $\langle n_{\text{max}} \rangle = t_S! \sigma^{t_S - 1}/Z_{t_S}^{(\text{poly})}$, where $Z_{t_S}^{(\text{poly})}$ is the partition function of polymerization (*SI Appendix, sections 8 and 9*). This probability is graphed as a function of $t_S$ and $\sigma$ in Fig. S4. At any fixed $t_S$, the probability of observing the maximer will tend to $1$ in the limit $\sigma \to \infty$. This puts a ceiling to $Q_{\text{max}}$ that is absent from the continuum description. In the $t_S$ dimension, the maximer probability decreases as $t_S$ increases at constant $\sigma$.

Polymerization as considered here has a natural analogy to bond percolation on a one-dimensional lattice. In the case of polymerization, the probability $p$ that any two protomers are linked by a bond is a function of $t_S$ and $\sigma$: $p = 1 - 2/\left(1 + \sqrt{1 + 4t_S\sigma} \right)$ (Fig. 5B and *SI Appendix, section 9*). The salient observation is that for small $t_S$, the maximer has a significant probability of already occurring at modest affinities: for $10$ protomers and $\sigma = 1$, $p$ is already $0.78$ and the maximer probability a reasonable $0.06$. For larger $t_S$, the maximer loses significance unless the affinity is scaled up correspondingly (*SI Appendix, section 10*). This is also reflected in the mass distribution (Fig. 5C).

Fig. 6A compares the discrete polymerizing-scaffold system with discrete multivalent scaffolds, much like Fig. 4A for the continuum case. The behavior of the discrete case is similar to that of the continuum case—with a few nuances that are prominent at low particle numbers and high affinities, such as the topmost orange curve. Its $\langle Q_{\text{poly}} \rangle$ profile does not hug the monovalent profile (bottom green chevron curve) to then increase its slope into the protozone peak as in the continuum case (Fig. 4A). A behavior like in the continuum case is observed for the lower orange and red curves, for which $\sigma$ is much weaker. In the continuum case, the affinity does not affect slope—the slope always changes from $1$ to $2$—but determines where that change occurs (Fig. 4C). The higher the affinity, the earlier the change. The topmost orange curve could be seen as realizing an extreme version of the continuum behavior in which an exceptionally high affinity causes a change to slope $2$ at unphysically low protomer concentrations. That such a scenario can be easily realized in the discrete case is due to the significant probability with which the maximer occurs at low particle numbers, similar to finite-size percolation. It bears emphasis that, as the number $t_S$ of protomers increases, the maximer probability decreases.

---

**Fig. 5.** Maximer. (A) The surface depicts the probability of observing the maximer as a function of $t_S$ and $\sigma$. (B) Here, the maximer probability is graphed as function of the probability $p(t_S, \sigma)$ that a bond exists between two protomers. Each curve corresponds to a particular $t_S$ with varying $\sigma$. $t_S$ ranges from $10$ (topmost curve) to $100$ (bottom curve) in increments of $10$, while $\sigma$ ranges from $1$ to $1,000$. (C) Mass distributions in the polymerizing-scaffold model. Any curve depicts the fraction of protomers in all length classes $n$, computed as $n! \sigma^{t_S - 1}/Z_{t_S}^{(\text{poly})}$ with $Z_{t_S}^{(\text{poly})}$ the partition function for polymerization with $t_S$ protomers (*SI Appendix, section 8*). Each curve corresponds to a given number of protomers: $t_S = 5$ (blue), $10$ (green), $15$ (plum), $20$ (red), $25$ (orange), $30$ (purple), $40$ (brown); affinity $\sigma = 3$ in all cases. When $t_S$ is small, the longest possible polymer—the “maximer”—is realized with appreciable frequency and dominates the mass distribution. As $t_S$ increases, at fixed $\sigma$, the maximal length class increases too, but its dominance fades.
(Fig. 5C), since the length of the maximer is $t_S$. However, once the maximer has receded in dominance, the increased number of length classes below it have gained occupancy and control the catalytic potential much like in the continuum case. Likewise, affinity does not appear to affect the slope of the downward leg as $t_S$ increases.

The discrete multivalent scaffold system behaves much like its continuum counterpart. In the affinity dimension (Fig. 6B), the discrete system shows a behavior similar to the continuum case with the qualification that $\langle Q_{\text{poly}} \rangle$ must level off to a constant, rather than increasing indefinitely. This is because, at constant $t_S$, an ever increasing affinity will eventually drive the system into its maximer ceiling. Because of the volume dependence of stochastic equilibrium constants, such an increase in affinity at constant protomer number can be achieved by any reduction of the effective reaction volume, for example, by confinement to a vesicle or localization to a membrane raft.

We determined SDs using stochastic simulations of the cases presented in Fig. 6A (SI Appendix, section 12). For a given $\langle Q \rangle$, the SD is larger after the prozone peak than before. Upon adding ligand-binding sites, the ratio of SD to mean (noise) increases much slower for the polymerizing system than for multivalent scaffolds.

**Main Conclusions**

Our theoretical analysis of a polymerizing-scaffold system shows that, at constant chemical potential, the system can be driven into criticality not only by increasing protomer concentration or affinity but by just increasing ligand concentrations. In equilibrium, polymerizing scaffolds exhibit a different type of prozone effect than multivalent ones: the polymerizing system provides a greater increase in catalytic potential on the upward leg above a certain protomer concentration $t_S$, delays the prozone peak, and significantly mitigates the collapse at high $t_S$. This behavior is caused by the response of the polymer-length distribution to changes in $t_S$ and could be exploited to identify the presence of a polymerizing scaffold in an experimental setting. When particles are present in discrete numbers, not continuous concentrations, system behavior is affected by the existence of a maximal polymer length. Behavior easily attainable at small protomer numbers requires extreme parameter values in the concentration-based description. Other than that, the discrete and the concentration-based systems behave similarly.

A polymerizing scaffold concentrates ligands locally and facilitates their interaction. In this it functions like a compartment, but through a mechanism that can be readily regulated by varying protomer concentration $t_S$ and polymerization affinity $\sigma$. We surmise that the regulation of catalytic potential in such systems is best modulated through the affinity $\sigma$, as this would not incur a prozone effect (Fig. 3A and Eq. 5) while being presumably faster and less costly than modulating $t_S$. Regulation of $\sigma$ could occur through posttranslational modification of the scaffold protomer. Regulation through $\sigma$ is most effective at a $t_S$ close to the prozone peak, whose location with respect to $t_S$ is robust to changes in $\sigma$ (Fig. 3B). Our analysis suggests that conformational constraints may not qualitatively subvert these observations, since adding a bond-distance constraint to ligand interactions did not fundamentally alter the combinatorial picture. Still, taking into account polymer conformation would increase realism. A significant extension of this work would consider scaffolding units of distinct types that form multiply interconnected aggregates. Such aggregates could control a larger diversity of ligand interactions and affect the aggregation/disaggregation transition.

**ACKNOWLEDGMENTS.** We gratefully acknowledge discussions with Tom Kolokotrones, Eric Deeds, and Daniel Merkle.

4. S. Ikeda et al., Axin, a negative regulator of the Wnt signaling pathway, forms a complex with GSK-3\(\beta\) and $\beta$-catenin and promotes GSK-3\(\beta\)-dependent phosphorylation of $\beta$-catenin. *EMBO J.* 17, 1371–1384 (1998).


12. C. Brangwynne et al., Germline p granules are liquid droplets that localize by controlled dissolution/condensation. Science 324, 1729–1732 (2009).


Supplementary Information for
Combinatorial protein-protein interactions on a polymerizing scaffold
Andrés Ortiz-Muñoz, Héctor F. Medina-Abarca and Walter Fontana

Walter Fontana.
E-mail: walter_fontana@hms.harvard.edu

This PDF file includes:
Supplementary text
Figs. S1 to S17
Table S1
References for SI reference citations

Contents
1 W and Q in the polymerizing scaffold model ................................................................. 3
2 Derivation of the general expression for the catalytic potential ................................................ 5
3 Overview of the polymerization system ............................................................................ 7
4 Mixtures of multivalent scaffolds .................................................................................... 9
5 Comparison between polymerizing and multivalent scaffold systems .................................. 10
6 Interaction horizon ........................................................................................................ 11
7 The discrete case ........................................................................................................... 13
   A Average catalytic potential ....................................................................................... 14
   B Boltzmann factor of a molecular species ................................................................ 14
   C Boltzmann factor of a state ................................................................................... 14
   D Degeneracy of a state ........................................................................................... 14
   E The partition function for a given resource vector ..................................................... 14
   F The average number of instances of a specific configuration in equilibrium ............... 15
8 Partition functions and average catalytic potential .............................................................. 15
   A Polymerizing scaffold without ligands ...................................................................... 15
   B Average catalytic potential of the polymerizing scaffold with ligands ....................... 16
   C Average catalytic potential of the multivalent scaffold with ligands ......................... 16
   D Remarks on numerical evaluation ............................................................................ 17
9 The maximer probability and 1D percolation ..................................................................... 17
10 Scaling behavior ........................................................................................................... 18
11 Unequal ligand concentrations and ligand binding affinities ............................................. 19
   A Polymerizing scaffold system .................................................................................. 19
   B Multivalent scaffold system .................................................................................... 19
12 Stochastic simulations ................................................................................................... 20
References ......................................................................................................................... 22

List of Figures
S1 Scaffold polymers and multivalent agents ........................................................................ 6
S2 The dependence of the length distribution on the protomer concentration ts and the affinity σ 7
S3 Concentrations within length classes ............................................................................. 8
S4 Mixtures of multivalent scaffolds .................................................................................... 9
S5 Polymerizing scaffold and multivalent scaffolds ............................................................ 10

www.pnas.org/cgi/doi/10.1073/pnas.1912745117
S6 Interaction horizon ............................................................................................................ 11
S7 Interaction horizon scenarios ................................................................................................. 12
S8 The impact of the interaction horizon ....................................................................................... 13
S9 Length distribution in continuum and discrete polymerization .................................................. 16
S10 Finite size 1D bond percolation and polymerization ................................................................. 18
S11 Scaling behavior of the maximer distribution .......................................................................... 18
S12 Effects in discrete and continuum polymerizing scaffold systems ............................................ 19
S13 Catalytic potential of multivalent scaffolds (discrete case) ....................................................... 20
S14 Catalytic potential of multivalent scaffolds (continuum case) ................................................... 20
S15 Stochastic simulations ........................................................................................................... 21
S16 Variance and noise .............................................................................................................. 21
S17 Distributions of catalytic potential ........................................................................................... 22

List of Tables

S1 List of symbols used in the main text and this Supplementary Information............................... 3
List of symbols used in the main text and this Supplementary Information

Table S1

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>ligand molecule of type $A$ (&quot;enzyme&quot;)</td>
</tr>
<tr>
<td>$B$</td>
<td>ligand molecule of type $B$ (&quot;substrate&quot;)</td>
</tr>
<tr>
<td>$S$</td>
<td>polymerizing scaffold protomer</td>
</tr>
<tr>
<td>$S_{(k)}$</td>
<td>multivalent scaffold with valency $k$ for each $A$ and $B$</td>
</tr>
<tr>
<td>$A_n$</td>
<td>Avogadro’s constant</td>
</tr>
<tr>
<td>$V$</td>
<td>system volume</td>
</tr>
<tr>
<td>$A_{pS_nB_q}$</td>
<td>a complex that is a member of a class consisting of $p$ $A$-molecules and $q$ $B$-molecules bound to an $S$-polymer of length $n$</td>
</tr>
<tr>
<td>${A_{pS_nB_q}}$</td>
<td>the class of complexes consisting of $p$ $A$-molecules and $q$ $B$-molecules bound to an $S$-polymer of length $n$</td>
</tr>
<tr>
<td>$a$</td>
<td>equilibrium concentration of free (unbound) $A$</td>
</tr>
<tr>
<td>$b$</td>
<td>equilibrium concentration of free (unbound) $B$</td>
</tr>
<tr>
<td>$s$</td>
<td>equilibrium concentration of free (unbound) $S$</td>
</tr>
<tr>
<td>$s_k$</td>
<td>equilibrium concentration of polymers of length $k$</td>
</tr>
<tr>
<td>$n$</td>
<td>for multivalent scaffold: number of binding sites per ligand type</td>
</tr>
<tr>
<td>$t_{A'}$</td>
<td>total concentration of $A$</td>
</tr>
<tr>
<td>$t_{B'}$</td>
<td>total concentration of $B$</td>
</tr>
<tr>
<td>$t_{S'}$</td>
<td>total concentration of $S$</td>
</tr>
<tr>
<td>$t_{S^{(k)}}$</td>
<td>total concentration of $k$-valent scaffold (non-polymeric)</td>
</tr>
<tr>
<td>$t_X$</td>
<td>total concentration of molecular species $X$</td>
</tr>
<tr>
<td>$t_{M}$</td>
<td>total number of binding sites (for $A$, or for $B$) in the system</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>equilibrium constant of $A$ binding to $S$</td>
</tr>
<tr>
<td>$\beta$</td>
<td>equilibrium constant of $B$ binding to $S$</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>equilibrium constant of $S$ binding to $S$ (polymerization)</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>generic affinity</td>
</tr>
<tr>
<td>$W(\cdot)$</td>
<td>energy-weighted generating function of molecular species</td>
</tr>
<tr>
<td>$Q$</td>
<td>catalytic potential: number of possible interactions between $A$-molecules and $B$-molecules on a scaffold species</td>
</tr>
<tr>
<td>$Q_{\text{dext}}$</td>
<td>catalytic potential of the direct interaction mechanism (Michaelis-Menten)</td>
</tr>
<tr>
<td>$Q_{\text{path}}$</td>
<td>catalytic potential of a multivalent scaffold system</td>
</tr>
<tr>
<td>$Q_{\text{poly}}$</td>
<td>catalytic potential of a polymerizing scaffold system</td>
</tr>
<tr>
<td>$Q_{\text{max}}$</td>
<td>maximum catalytic potential of a system</td>
</tr>
<tr>
<td>$q_{\alpha x}(i, h)$</td>
<td>maximal catalytic potential for a polymeric scaffold of length $i$ for ligands that can interact when separated by at most $h$ polymer bonds</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>“catalytic horizon”: max distance within which two ligands can interact</td>
</tr>
<tr>
<td>$p(t_{M}, t_X, \gamma)$</td>
<td>probability that one among $t_{M}$ scaffold sites is occupied with a ligand of type $X$ at concentration $t_X$ and binding affinity $\gamma$</td>
</tr>
<tr>
<td>$\bar{n}$</td>
<td>a system state $(n_1, n_2, \ldots, n_C)$, where $n_i$ is the number of particles of species $Y_i$</td>
</tr>
<tr>
<td>$\bar{\bar{i}}$</td>
<td>$i$-th generic species</td>
</tr>
<tr>
<td>$d(\bar{n})$</td>
<td>degeneracy of state $\bar{n}$</td>
</tr>
<tr>
<td>$Y_i$</td>
<td>generic molecular species</td>
</tr>
<tr>
<td>$X_i$</td>
<td>“atomic” building block in the assembly of $Y_i$</td>
</tr>
<tr>
<td>$T$</td>
<td>number of &quot;atom&quot; types available for assembly (here $T = 3$)</td>
</tr>
<tr>
<td>$\omega_i$</td>
<td>number of symmetries (graph automorphisms) of the $i$-th generic species $Y_i$</td>
</tr>
<tr>
<td>$K_T$</td>
<td>rate constant of the $T$th reaction</td>
</tr>
<tr>
<td>$\mathcal{P}$</td>
<td>assembly path</td>
</tr>
<tr>
<td>$\varepsilon_i$</td>
<td>free energy of formation of $Y_i$</td>
</tr>
<tr>
<td>$\varphi(\bar{i}, Y_i)$</td>
<td>the number of distinct ways of realizing a single instance of $Y_i$ given resources $\bar{i}$</td>
</tr>
<tr>
<td>$\xi$</td>
<td>rescaling factor</td>
</tr>
<tr>
<td>$\langle \cdot \rangle$</td>
<td>expectation value</td>
</tr>
<tr>
<td>$Z$</td>
<td>partition function</td>
</tr>
</tbody>
</table>

Supporting Information Text

1. $W$ and $Q$ in the polymerizing scaffold model

In this section we step through the treatment of the polymerizing scaffold model with more granularity. A polymerizing scaffold protomer $S$ has $1$ binding site for each ligand $A$ and $B$. Let $\{A_{pS_nB_q}\}$ be the set of complexes (configurations) consisting of a scaffold polymer with $n$ protomers, $p$ agents of type $A$ and $q$ agents of type $B$; let $\{\{A_{pS_nB_q}\}\}$ denote their aggregate equilibrium concentration. The equilibrium concentration of any particular representative $A_{pS_nB_q}$ of that class is given by

$$[A_{pS_nB_q}] = \sigma^{n-1} a^p b^q s^n a^p b^q = \sigma^{n-1} s^n (\alpha a)^p (\beta b)^q,$$

where $a$, $b$, $s$ are the equilibrium concentrations of free $A$, $B$, and $S$, respectively; $\alpha$ denotes the equilibrium constant of $A$ binding to $S$ and, similarly, $\beta$ and $\sigma$ are the equilibrium constants for $B$ binding to $S$ and for $S$ binding to $S$, respectively. All
binding interactions are posited to be mechanistically independent of one another.

In an equilibrium treatment, a system of reactions only serves to define a set of reachable complexes and could be replaced with any other mechanism, no matter how unrealistic, as long as it produces the same set of reachable configurations. Hence we could posit that a polymer of length n is generated by a reversible “reaction” in which all constituent protomers come together at once. The equilibrium constant of such an imaginary reaction must be the exponential of the energy content of a polymer of length n, which in our case is simply (n−1) times the energy content of a single bond, i.e. ln σ. Thus, the equilibrium constant of the fictitious one-step assembly reaction is σn−1 and [1] follows.

To aggregate the equilibrium concentrations of all molecular configurations in the class \( \{ A_p S_n B_q \} \) we note that the set \( \{ A_p S_n B_q \} \) includes \( \binom{n}{p} \binom{m}{q} \) configurations with the same energy content \( \sigma^{n−1} \alpha^{p} \beta^{q} \). Summing over all \( p \) and \( q \), yields the contribution of the polymer length class \( n \), \( \{ A_n S_n B_n \} \)

\[
[(A_n S_n B_n)] = \sigma^{n−1} s^n \left[ \sum_{p=1}^{n} \binom{n}{p} \alpha^p \beta^n \right] \left[ \sum_{q=1}^{n} \binom{n}{q} \beta^q \right] = \sigma^{n−1} s^n (1 + \alpha a) (1 + \beta b)^n = \frac{1}{\sigma} (\sigma s (1 + \alpha a)(1 + \beta b))^n
\]  

[2]

Summing over all equilibrium concentrations defines a function \( W \):

\[
W = a + b + \frac{1}{\sigma} \sum_{n=1}^{\infty} (\sigma s (1 + \alpha a)(1 + \beta b))^n = a + b + (1 + \alpha a)(1 + \beta b) \sum_{n=0}^{\infty} (\sigma s (1 + \alpha a)(1 + \beta b))^n
\]  

[3]

When viewing \( a, b \) and \( s \) as formal variables, \( W \) acts as a generating function of energy-weighted configurational counts. By differentiating \( W \) with respect to \( s \) each \( s \)-containing term gets multiplied with the exponent of \( s \), which is the \( S \)-content of the respective configuration. Multiplying by \( s \) then restores the exponent and recovers the equilibrium concentration of the respective configuration. Summing over all configurations so treated, yields the total amount of \( S \) protomers in the system and thus a conservation relation. This holds for all formal variables representing the “atoms”, or building blocks, of the system:

\[
t_A = a \frac{\partial W(a, b, s)}{\partial a}, \quad t_B = b \frac{\partial W(a, b, s)}{\partial b}, \quad t_S = s \frac{\partial W(a, b, s)}{\partial s}.
\]  

[4]

By solving the equations [4], we obtain the equilibrium concentrations of free \( A, B \), and \( S \) needed to compute the equilibrium concentration of any configuration:

\[
a = \frac{\alpha t_A - \alpha t_S - 1 + \sqrt{(\alpha t_A + \alpha t_S + 1)^2 - 4\alpha^2 t_A t_S}}{2\alpha}
\]  

[5]

\[
b = \frac{\beta t_B - \beta t_S - 1 + \sqrt{(\beta t_B + \beta t_S + 1)^2 - 4\beta^2 t_B t_S}}{2\beta}
\]  

[6]

\[
s = \frac{2}{\sigma^2 t_S} \left[ \frac{2\sigma t_S + 1 - \sqrt{\sigma t_S + 1}}{\alpha t_A - \alpha t_S + 1 + \sqrt{(\alpha t_A + \alpha t_S + 1)^2 - 4\alpha^2 t_A t_S}} \right] \left[ \beta t_B - \beta t_S + 1 + \sqrt{(\beta t_B + \beta t_S + 1)^2 - 4\beta^2 t_B t_S} \right]
\]  

[7]

Carrying out the geometric sum in [3] yields Eq. [3] in the main text:

\[
W(a, b, s) = a + b + \frac{s(1 + \alpha a)(1 + \beta b)}{1 - \sigma s(1 + \alpha a)(1 + \beta b)}.
\]  

[8]

The same manipulation of \( W \) used to obtain [4] can be carried out twice, once for \( a \) and once for \( b \), to yield the catalytic potential of the system:

\[
Q = ab \frac{\partial^2 W(a, b, s)}{\partial a \partial b},
\]  

[9]

given as Eq. [4] in the main text.

By setting \( a = b = 0 \), we recover the standalone polymerization system with

\[
W(s) = \frac{s}{1 - \sigma s}
\]  

[10]

and \( s \) obtained from solving \( t_S = dW(s)/ds \):

\[
s = \frac{1}{4\sigma} \left( \sqrt{4 + \frac{1}{\sigma s}} - \sqrt{\frac{1}{\sigma s}} \right)^2.
\]  

[11]

as in Eq. [2] of the main text. We discuss the main properties of the standalone polymerization system in section 3 of this Appendix. In an equilibrium setting, the critical point of the model with ligands \( A \) and \( B \) should be the same as that of the polymerization system without ligands, namely \( t_S \to \infty \) or \( \sigma \to \infty \). This is not obvious from \( W \) (whose critical point \( Q \) inherits) as given in [8] with solutions [5]-[7]. However, it is made explicit in an alternative, more insightful derivation of the equilibrium catalytic potential \( Q \) given in section 2 of this Appendix.
2. Derivation of the general expression for the catalytic potential

In this section we derive Eq. [5] of the main text.

We consider a multivalent scaffold agent $S$ with $n_A$ binding sites for $A$ and $n_B$ binding sites for $B$. Our goal is to calculate the catalytic potential $Q_{\text{multi}}$ of a system consisting of $A$-agents at concentration $t_A$, $B$-agents at concentration $t_B$, and $S$-agents at concentration $t_S$.

The function $W(a, b, s)$, introduced in the main text for the polymerizing scaffold system, sums up the equilibrium concentrations of all possible entities in the system. The same concept applies to a multivalent scaffold:

$$W_{\text{multi}}(a, b, s) = a + b + s(1 + a\alpha)^n_A (1 + b\beta)^n_B$$ \[12\]

with $a$, $b$, and $s$ the equilibrium concentrations of the free $A$, $B$, and $S$, respectively. The catalytic potential $Q_{\text{multi}}$ of the multivalent scaffold system is

$$Q_{\text{multi}} = a b \frac{\partial^2}{\partial a \partial b} W_{\text{multi}}(a, b, s) = s \alpha a b n_A n_B (1 + a\alpha)^{n_A-1} (1 + b\beta)^{n_B-1}.$$ \[13\]

The equilibrium concentrations $a$, $b$, and $s$ are determined by the system of conservation equations

$$a \frac{\partial}{\partial a} W = t_A, \quad b \frac{\partial}{\partial b} W = t_B, \quad s \frac{\partial}{\partial s} W = t_S.$$ \[14\]

However, we can bypass solving these equations by calculating the concentrations directly, which serendipitously gives us an intelligible expression for the catalytic potential $Q$ in general.

We first calculate the equilibrium concentration of the fully occupied scaffold configuration, $[A_{n_A}S B_{n_B}]$ by reasoning at the level of binding sites. The concentration of sites available for binding to $S$ are denoted by $a$, which is also the concentration of free $A$-agents. Since each $A$-binding site on $S$ is independent, the equilibrium fraction of $S$-agents that are fully occupied with $A$-agents is simply

$$\frac{[\{A_{n_A}S\}]}{t_S} = \left(\frac{aa}{1 + a\alpha}\right)^{n_A}.$$ \[15\]

The expression in parentheses is the single-site binding equilibrium. Likewise, let $[\alpha s]$ be the concentration of free $A$-binding sites on $S$-agents and $[as]$ the concentration of bonds between $A$- and $S$-agents. In equilibrium we have that

$$aa[s] = [as], \quad n_A t_S = [s] + [as], \quad t_A = a + [as].$$ \[16\]

Hence, $a = [as]/(a[s])$ or $a = (t_A - a)/(a[s]) = (t_A - a)/(a(n_A t_S - t_A + a))$, which yields a quadratic in $a$ whose solution is

$$a = \frac{1}{2a}(\alpha t_A - n_A a t_S - 1 + \sqrt{(\alpha t_A - n_A a t_S - 1)^2 + 4\alpha t_A}).$$ \[17\]

We plug [17] into [15] to obtain

$$\frac{[\{A_{n_A}S\}]}{t_S} = \left(\frac{\alpha t_A - n_A a t_S - 1 + \sqrt{(\alpha t_A - n_A a t_S - 1)^2 + 4\alpha t_A}}{\alpha t_A - n_A a t_S + 1 + \sqrt{(\alpha t_A - n_A a t_S - 1)^2 + 4\alpha t_A}}\right)^{n_A}.$$ \[18\]

The same reasoning holds for the (independent) binding of $B$ to $S$:

$$\frac{[\{SB_{n_B}\}]}{t_S} = \left(\frac{\beta t_B - n_B \beta t_S - 1 + \sqrt{(\beta t_B - n_B \beta t_S - 1)^2 + 4\beta t_B}}{\beta t_B - n_B \beta t_S + 1 + \sqrt{(\beta t_B - n_B \beta t_S - 1)^2 + 4\beta t_B}}\right)^{n_B}.$$ \[19\]

At this point it is useful to abbreviate

$$a_\pm = a_\pm(t_A, t_S, \alpha, n_A) = \alpha t_A - n_A a t_S \pm 1 + \sqrt{(\alpha t_A - n_A a t_S - 1)^2 + 4\alpha t_A}$$ \[20\]

and

$$b_\pm = b_\pm(t_B, t_S, \beta, n_B) = \beta t_B - n_B \beta t_S \pm 1 + \sqrt{(\beta t_B - n_B \beta t_S - 1)^2 + 4\beta t_B}$$

Note that these abbreviations are dimensionless functions of the parameters $t_A$, $t_S$, $\alpha$ and $n_A/B$. Because $A$ and $B$ bind independently, we can combine [18] and [19] to obtain:

$$[A_{n_A}S B_{n_B}] = t_S \frac{a_+^{n_A} b_+^{n_B}}{a_-^{n_A} b_-^{n_B}} = (aa)^{n_A} (\beta b)^{n_B} s.$$ \[21\]

where the last equation is the equilibrium concentration in terms of free $A$, free $B$, and free $S$, as mentioned in the Introduction of the main text (and section 1 of this Appendix). The expression $a$ for free $A$ is given by [17], or $a = a_-/(2\alpha)$. The expression $b$ for free $B$ is analogous, $b = b_-/(2\beta)$. Equation [21] now yields $s$:

$$s = t_S \frac{1}{(aa)^{n_A} (\beta b)^{n_B}} \frac{a_+^{n_A} b_+^{n_B}}{a_-^{n_A} b_-^{n_B}} = t_S \frac{2^{n_A} 2^{n_B}}{a_+^{n_A} b_+^{n_B}}.$$ \[22\]
To summarize, using abbreviations [20]:

\[ a = \frac{a}{2\alpha}, \quad b = \frac{b}{2\beta}, \quad s = t_S \left( \frac{2}{a_+} \right)^{n_A} \left( \frac{2}{b_+} \right)^{n_B}. \]  

[23]

Keep in mind that \(a_{+/-}\) and \(b_{+/-}\) are not constants, but functions of the system parameters. We now insert [23] into [13] to obtain

\[ Q_{\text{multi}} = n_A n_B t_s s \frac{(\frac{\alpha a}{1 + \alpha a}) (\frac{\beta b}{1 + \beta b}) (1 + \alpha a)^{n_A} (1 + \beta b)^{n_B}}{2} \]

[24]

\[ = n_A n_B t_s \left( \frac{2}{a_+} \right)^{n_A} \left( \frac{2}{b_+} \right)^{n_B} \frac{(\alpha a)}{1 + \alpha a} \frac{(\beta b)}{1 + \beta b} (1 + \alpha a)^{n_A} (1 + \beta b)^{n_B} \]

\[ = n_A n_B t_s \left( \frac{\alpha a}{1 + \alpha a} \right) \frac{(\beta b)}{1 + \beta b} \frac{a_- b_+}{a_+ b_+}. \]  

[25]

The cancellations are due to \(2\alpha a = a_-\) (from [23]) and \(a_+ = a_- + 2\) (from [20]).

Return to equation [18] and set \(n_A = 1\). This gives the fraction of \(A\)-binding sites (of monovalent scaffold agents) that are occupied, that is, the probability that an \(A\) is bound:

\[ p(t_s, t_A, s) = \frac{a_{-}(t_A, t_S, s, s, 1)}{a_{+}(t_A, t_S, s, s, 1)} = \frac{\alpha t_A - \alpha t_S - 1 + \sqrt{(\alpha t_A - \alpha t_S - 1)^2 + 4\alpha t_A}}{\alpha t_A - \alpha t_S + 1 + \sqrt{(\alpha t_A - \alpha t_S - 1)^2 + 4\alpha t_A}}. \]  

[26]

In the site-oriented view it does not matter whether an \(A\)-binding site belongs to a monovalent scaffold agent or to an \(n\)-valent scaffold agent. At the same agent concentration \(t_s\), the \(n\)-valent agent simply provides \(n\) times more sites. Thus, the probability that an \(A\) is bound if the scaffolds are \(n\)-valent is

\[ p(n t_S, t_A, s) = \frac{a_{-}(t_A, n t_S, s, s, 1)}{a_{+}(t_A, n t_S, s, s, 1)} = \frac{a_{-}(t_A, n t_S, s, s, 1)}{a_{+}(t_A, n t_S, s, s, 1)}. \]  

[27]

since the number of binding sites only scales \(t_s\) in [20]. With these observations, we can rephrase [24] as the product of two terms:

\[ Q_{\text{multi}} = p(n t_s, t_A, s) p(n t_s, t_B, s) n_A n_B t_s. \]

Term (I) is the probability that a site of some \(S\) is occupied by \(A\) and a site of some \(S\) is occupied by \(B\). Term (II) counts the maximal number of possible interactions between \(A\) and \(B\) agents in the system.

**Fig. S1.** A multivalent scaffold agent can be thought as representing a particular scaffold polymer configuration.

Let \(S_{(i)}\) denote an agent of valency \(i\) for both ligands and let \(t_{S_{(i)}}\) denote its concentration. In a mixture of multivalent scaffold types of distinct valencies \(i = 1, \ldots, n\) present at concentrations \(t_{S_{(i)}}\), the catalytic potentials of each type add up to that of the mixture, \(Q_{\text{mix}}\):

\[ Q_{\text{mix}} = p \left( \sum_{i=1}^{n} i t_{S_{(i)}}, t_A, s \right) p \left( \sum_{i=1}^{n} i t_{S_{(i)}}, t_B, s \right) \sum_{i=1}^{n} i^2 t_{S_{(i)}}. \]  

[28]
Generally, we can write $Q_{\text{mix}}$ as

$$Q_{\text{mix}} = p(t_{\text{sit}}, t_A, \alpha)p(t_{\text{sit}}, t_B, \beta) Q_{\text{max}}(\vec{t}_S). \quad [29]$$

In [29], $t_{\text{sit}}$ is the total concentration of binding sites, regardless of how they are partitioned across scaffold agents, $\vec{t}_S = (t_{S(1)}, \ldots, t_{S(n)})$ is a partition of sites across scaffold molecules of different valencies, and $Q_{\text{max}}$ is the maximal attainable number of enzyme-substrate interactions in the system, which depends on the concentration of scaffolds and their valency.

If the mixture results from a polymerization process between monovalent scaffolds $S \equiv S(1)$, we identify a polymer of length $l$ with an $l$-valent scaffold agent (Figure S1).

The concentrations $t_{S(i)}$ are endogenously determined by polymerization at equilibrium:

$$t_{S(i)} = \sigma^{i-1} s_i,$$

where the expression for $s$ is given by the expression for the equilibrium concentration of free monomer in the polymerization system absent ligands, expression [11] in section 1 (Eq. [2] in the main text). Using these $t_{S(i)}$ in the sum [28], which in the continuum case runs to $n = \infty$, yields the Eq. [6] for $Q_{\text{poly}}$ in the main text:

$$Q_{\text{poly}} = p(t_S, t_A, \alpha)p(t_S, t_B, \beta) \sum_{n=1}^{\infty} n^2 \sigma^{n-1} s^n = p(t_S, t_A, \alpha)p(t_S, t_B, \beta) \frac{(1 + \sigma s)}{(1 - \sigma s)}, \quad [30]$$

with $p(\cdots)$ given by [25].

3. Overview of the polymerization system

---

Fig. S2. The dependence of the length distribution on the protomer concentration $t_S$ and the affinity $\sigma$. A: The curves depict the length distribution $s_i$ of the linear polymerization subsystem with varying $t_S$ at $\sigma = 10^8 \text{ M}^{-1}$. Blue: $t_S = 2 \cdot 10^{-7} \text{ M}$, orange: $t_S = 4 \cdot 10^{-7} \text{ M}$, green: $t_S = 6 \cdot 10^{-7} \text{ M}$, red: $t_S = 8 \cdot 10^{-7} \text{ M}$, purple: $t_S = 1 \cdot 10^{-6} \text{ M}$. The inset plots the same curves in lin-log. B: The curves depict the concentrations of protomers in each length class, that is, the "mass" distribution of $s_i$, under the same conditions as in panel A. C: The curves depict the length distribution $s_i$ with varying polymerization affinity $\sigma$ at $t_S = 6 \cdot 10^{-8} \text{ M}$. Blue: $\sigma = 10^5 \text{ M}^{-1}$, orange: $\sigma = 10^7 \text{ M}^{-1}$, green: $\sigma = 10^8 \text{ M}^{-1}$, red: $\sigma = 10^{10} \text{ M}^{-1}$, purple: $\sigma = 10^{12} \text{ M}^{-1}$, brown: $\sigma = 10^{13} \text{ M}^{-1}$, light blue: $\sigma = 10^{14} \text{ M}^{-1}$, D: As in panel B, but with varying affinity $\sigma$ (as in panel C) at $t_S = 6 \cdot 10^{-8} \text{ M}$. For all panels $\alpha = \beta = 10^7 \text{ M}^{-1}$, $t_A = 15 \cdot 10^{-9} \text{ M}$ and $t_B = 5 \cdot 10^{-7} \text{ M}$. 

In this section we summarize some combinatorial properties of the polymerization subsystem. Understanding the concentration profile of the polymer length distribution is useful for rationalizing the overall behavior with respect to catalytic potential, because we can view the polymerizing scaffold system as a mixture of multivalent scaffolds whose concentration is set by
polymerization. Since this is the simplest conceivable polymerization system, it would surprise us if anything being said here isn’t already known in some form or another. Some of the features described can be found in Flory (1).

Let \( S_n \) be a polymer of length \( n \) and let \( s_n \) denote the equilibrium concentration of polymers in length class \( n \). To conform with our previous notation, we shall refer to the equilibrium concentration of the monomer as \( s \equiv s_1 \) and to the monomer species as \( S \equiv S_1 \). As stated repeatedly,

\[
    s_n = \sigma^{n-1} s_1^n \quad \text{with} \quad s = \frac{1}{4\sigma} \left( \sqrt{4 + \frac{1}{\sigma \tau_s}} - \frac{1}{\sigma \tau_S} \right)^2
\]  

[31]

Figure S2 shows the dependency of \( s_n \) on the total protomer concentration \( t_S \) (panels A and B) and the affinity \( \sigma \) (panels C and D). Obviously, \( s_n \) is a geometric progression, thus linear in a lin-log plot for all parameter values (insets of panel A and C).

In the \( t_S \) dimension, \( s_n \) approaches \( 1/\sigma \) from below for each \( n \) and there is no value of \( t_S \) that maximizes \( s_n \). In the \( \sigma \) dimension, \( s_n \) approaches \( 0 \) like \( 1/\sigma \) (in the lin-log plot, inset of panel C, the straight lines become less tilted and sink toward \( 0 \)); see also expansions [36] and [37] below. However, for any given length class \( n \), there is a \( \sigma \) that maximizes the concentration of that class:

\[
    \sigma = \frac{n^2 - 1}{4t_S}.
\]  

[32]

At that \( \sigma \), the respective \( s_n \) is the most frequent, i.e. the most dominant, length class. It does not mean that \( s_n \) is at its most frequent, for \( s_n \) rises to \( 1/\sigma \) as \( t_S \to \infty \). In the continuum description, the most frequent polymer class is always the monomer, for any \( t_S \) or \( \sigma \). This is much more pronounced in the \( t_S \) dimension than the \( \sigma \) dimension.

![Fig. S3. Concentrations within length classes.](image)

Fig. S3. Concentrations within length classes. These panels are complementary to those in Figure S2. Each curve tracks the concentration of a particular length class \( n \) as protomer concentration \( t_S \) and affinity \( \sigma \) are varied, effectively following the changes along a vertical cut across the curves in Figure S2. Blue: \( n = 1 \), orange: \( n = 2 \), green: \( n = 3 \), red: \( n = 5 \), purple: \( n = 10 \). All other parameters as in Figure S2. A: Concentration \( s_n \) of length class \( n \) with varying \( t_S \). B: Concentration \( n_s \) of the mass in length class \( n \) with varying \( t_S \). Panel C: Concentration \( s_n \) of length class \( n \) with varying \( \sigma \). Panel D: Concentration \( n_s \) of the mass in length class \( n \) with varying \( \sigma \).

Panels B and D of Figure S2 show the “mass” distribution, \( n_s \), i.e. the concentration of protomers in each length class. For all values of \( t_S \) and \( \sigma \) the mass exhibits a maximum at some class length. This maximum wanders towards ever larger \( n \) with increasing \( t_S \) and \( \sigma \), while its value steadily increases with \( t_S \), whereas it decreases with increasing \( \sigma \). The length class \( n \) whose mass is maximized at a given \( t_S \) and \( \sigma \) is

\[
    n_{\text{max}} = \left[ \log \left( \frac{4t_S \sigma}{(\sqrt{1 + 4t_S \sigma} - 1)^2} \right) \right]^{-1},
\]  

[33]
and, for given $\sigma$ and $n$, the $t_S$ at which the class $n$ becomes the most massive of all classes is given by

$$t_S = \frac{\exp(1/n)}{\sigma(1 - 2\exp(1/n) + \exp(2/n))}.$$ \[34\]

The pink squares on the blue multivalent scaffold curves in Figure 4B of the main text correspond to the catalytic potential $Q$ that obtains at this concentration of sites. The same expression obtains for $\sigma$ by swapping $t_S$ and $\sigma$. At the $t_S$ at which the mass in class $n$ peaks, the concentration of the class is

$$s_{n_{\text{max}}} = \frac{1}{n\sigma}.$$ \[35\]

independent of $n_{\text{max}}$. Equation [33] assumes a continuous $n$; thus, to account for the discrete nature of polymer length, the actual $n_{\text{max}}$ should be the nearest integer to the $n_{\text{max}}$ given in [33]. Accordingly, the actual value of $s_{n_{\text{max}}}$ in expression [35] will wobble slightly.

Switching perspective from the length distribution to the behavior within a length class yields Figure S3. The expansion of $s_n$ shows how each length class approaches its limit as $t_S \to \infty$ or $\sigma \to \infty$ (multiply by $n$ for the mass distribution):

As $t_S \to \infty$, $s_n \to \frac{1}{\sigma} \left(1 - \frac{n}{\sigma^{1/2} t_S^{1/2}} + O\left(\frac{1}{t_S}\right)\right)$ \[36\]

As $\sigma \to \infty$, $s_n \to 0$ with $\frac{1}{\sigma} \left(1 - \frac{n}{\sigma^{1/2} t_S^{1/2}} + O\left(\frac{1}{\sigma^2}\right)\right)$ \[37\]

4. Mixtures of multivalent scaffolds

Figure S4A shows the $Q_{\text{mix}}$-surface [28] of a bivalent and trivalent scaffold mixture. The main observation is the asymmetry in the effect on $Q$ upon adding $S_{(1)}$ to a fixed amount of $S_{(2)}$ compared to the other way around—blue versus red mesh lines in Figure S4. Upon adding $S_{(3)}$, the ligands $A$ and $B$ re-equilibrate over the available binding sites. Over a range of $[S_{(2)}]$, this equilibration is more likely to result in $A$ and $B$ agents ending up on the same $S_{(3)}$ scaffold than on the same $S_{(2)}$ scaffold. This is most pronounced at small $S_{(2)}$ and disappears gradually as the addition of binding sites drives the system past the prozone peak due to the $p^2$ term in [28]. The orange curve shows the $Q$-profile of a mixture in which $S_{(3)}$ and $S_{(2)}$ are increased in equal amounts. The dotted curves are the projections of the mixture curve on each component axis for the purpose of comparison with the $Q$-curves of each component in isolation. This behavior is more dramatic in binary mixtures of multivalent scaffolds with large valency differences (Figure S4B).

Fig. S4. Mixtures of multivalent scaffolds. A: The graphics renders the $Q_{\text{mix}}$-surface of a mixture of a bivalent and trivalent scaffold. The orange line is the $Q$-profile when both agents are added in equal amounts to the mix. The dotted lines are projections of the orange line for comparison with the homogeneous scaffold systems. B: Same as in panel A but for a mixture of $S_{(2)}$ and $S_{(30)}$; only the portion of the surface at low scaffold concentrations is shown. The green curve shows the $Q$-trajectory for the binary mixture that would obtain when $[S_{(2)}]$ and $[S_{(30)}]$ are set by the polymerizing scaffold system with increasing $t_S$. The green curve is the whole trajectory, because both $[S_{(2)}]$ and $[S_{(30)}]$ converge to $1/\sigma = 10^{-8}$ M (Figure S3). Other parameters: $\alpha = \beta = 10^7$ M$^{-1}$, $t_A = 15 \cdot 10^{-9}$ M$^{-1}$, $t_B = 5 \cdot 10^{-7}$ M.

In a polymerizing scaffold system, the concentrations $s_i \equiv [S_{(i)}]$ and $s_j \equiv [S_{(j)}]$ do not increase in equal amounts when $t_S$ is increased, but are related by a factor $(\sigma s)^{1-\beta}$. Since $\sigma s < 1$ for $t_S < \infty$, there is a lag between the rise of $S_{(i)}$ and $S_{(j)}$, where $S_{(i)}$ increases before $S_{(j)}$ for $i < j$; this lag is more dramatic the bigger the difference $|i - j|$ (Figure S4B, green curve). In the polymerizing system, as $t_S$ increases, the ratio of $S_{(i)}$ and $S_{(j)}$ will tend to 1, but by then the between-class prozone is taking its toll. In sum, the “stealing” of ligands by higher length classes from lower ones is the reason for the turn towards a steeper
slope of $Q_{\text{poly}}$ at $t_S$ values at which polymerization becomes effective (Figure 4A in the main text). Incidentally, the shift of ligands from lower towards higher valency classes also tends to flatten the intrinsic slope of the downward leg of lower valency classes after the prozone peak, contributing further to prozone mitigation in the overall system.

5. Comparison between polymerizing and multivalent scaffold systems

In the main text, Figure 4A and 4B, we compare multivalent scaffolds with the polymerizing scaffold system. Figure S5 places that comparison in the context of the full $Q_{\text{poly}}$ surface to show the effectiveness of regulating the affinity $\sigma$.

While even for $n_A = n_B = n$ and $\alpha = \beta$, $Q_{\text{multi}}$ is a cumbersome expression, determining the concentration of scaffold agents $t_S$ for which $dQ_{\text{multi}}/dt_S = 0$ yields a simple solution

$$t_S = \frac{1}{n} \left( \frac{1}{\alpha} + \frac{t_A + t_B}{2} \right).$$

Equation [38] shows that when plotting $Q_{\text{multi}}$ against the concentration of sites $t_{\text{sit}} = nt_S$, as in Figure S5 and Figure 4A of the main text, the prozone peaks line up for all valencies $n$.

$$Q_{\text{multi}} = \frac{\alpha t_A \beta t_B}{1 + \alpha t_A + \beta t_B + \alpha \beta t_A t_B} n^2 t_S + O(t_S^3).$$

Expanding $Q_{\text{multi}}$ (assuming $n_A = n_B = n$) in $t_S$ near zero, yields

Hence in a log-log plot, the up-leg of $Q_{\text{multi}}(n)$ has, to leading order, slope 1 and offset $n$ when plotted against sites $t_{\text{sit}} = nt_S$ as in Figure 4A of the main text. Similarly, expanding $Q_{\text{multi}}$ in $t_S$ near infinity, yields

$$Q_{\text{multi}} = t_A t_B \frac{1}{t_S} + O(1/t_S^3),$$

and hence, to leading order, a slope of $-1$ in a log-log plot in the down-leg after the prozone peak and an offset of $n$ when plotted against $t_{\text{sit}}$ as in Figure 4A of the main text.

The expansion of $Q_{\text{poly}}$ in $t_S$ ($= t_{\text{sit}}$) around zero yields

$$Q_{\text{poly}} = \frac{\alpha t_A \beta t_B}{1 + \alpha t_A + \beta t_B + \alpha \beta t_A t_B} t_S + \left[ f(\alpha, \beta, t_A, t_B) + g(\alpha, \beta, t_A, t_B) \sigma \right] t_S^2 + O(t_S^3)$$

with $f()$ and $g()$ functions of the indicated parameters. The leading-order term is the same as the $Q_{\text{multi}}$ of the monovalent scaffold, and is independent of $\sigma$, which enters the second-order term. Accordingly, for small $t_S$, $Q_{\text{poly}}$ hugs the $Q$ of the monovalent scaffold as if there was no polymerization; as $t_S$ increases, $\sigma$ (i.e., polymerization) becomes effective and $Q_{\text{poly}}$ doubles its slope upward. This is clearly seen in Figure 4A of the main text. Some microscopic consequences from building up a length distribution as $t_S$ increase are discussed in section 4.

Expanding $Q_{\text{poly}}$ in $t_S$ at infinity yields

$$Q_{\text{poly}} = 2t_A t_B \sqrt{\sigma} \sqrt{\frac{1}{t_S}} + O(1/t_S^{3/2}),$$

where the $p(t_S, t_A, \alpha)p(t_S, t_B, \beta)$ component scales with $t_A t_B/t_S^2$ and the $Q_{\text{max}}$ component with $2t_S^{3/2} \sqrt{\sigma}$ to leading order. As a result, the slope of the down-leg of $Q_{\text{poly}}$ after the prozone peak in a log-log plot is $-1/2$. 
6. Interaction horizon

Structural constraints might prevent every catalyst $A$ on a polymeric scaffold from interacting with all substrates $B$ bound to the same polymer. To obtain a rough sense of how such constraints could impact the catalytic potential $Q$, we define an “interaction horizon”, $h$, Figure S6. The horizon $h$ is the farthest distance in terms of scaffold bonds that a bound $A$ can “reach”. This means that a given bound enzyme $A$ can interact with at most $2h+1$ substrate agents $B$: $h$ to its “left”, $h$ to its “right” and one bound to the same protomer, Figure S6A. For example, in Figure S6B, the 2-horizon of the $A$ at position 1 includes the $B$s at positions 2 and 3, but not at position 5. Likewise, the $B$ at position 2 is outside the 2-horizon of the $A$ at position 5, whereas all $B$s are within reach of the $A$ at position 3. Clearly, the interaction horizon only modulates the $Q_{max}$ in equation [29] of a polymer of length $n$: more precisely, it modulates the interaction factor—the $n^2$ in the first equation of [30]. We now write this factor as $q_{max}(n, h)$; it replaces the $n^2$ in [30].

![Interaction horizon schematic](image)

**Fig. S6.** Interaction horizon. The schematic illustrates the case in which the horizon $h$ is less than the polymer length $n$. In this case, each $A$-binding position can interact with at most $h$ $B$-binding positions on its “left” or “right” side. When $h \geq n$, every $A$-position can interact with every $B$-position.

To reason about the catalytic combinations, we first consider the case $0 \leq h \leq \lfloor n/2 \rfloor$:

$$q_{max}(n, h) = (n - 2h)(2h + 1) + 2h(h + 1) + 2 \sum_{k=1}^{h-1} (h - k) = n(2h + 1) - h(h + 1)$$ \[43\]

Term I refers to the $n - 2h$ positions in the middle region of the chain that can interact with the full complement of $2h + 1$ sites within its horizon. Term II refers to the $h$ positions at each end of the chain and accounts for all $h + 1$ sites reachable towards the interior of the chain. Term III accounts for the remaining $h - k$ locations towards the end of the chain that can be reached from a position considered in term II; these locations depend on that position’s distance $k$ from the end of the chain. For $\lfloor n/2 \rfloor < h \leq n - 1$ we obtain

$$q_{max}(n, h) = (2h - n)n + 2(n - h)(h + 1) + 2 \sum_{k=1}^{n-h} (k - 1) = n(2h + 1) - h(h + 1)$$ \[44\]

In analogy to [43], Term $\Gamma$ refers to the $2h - n$ positions that can access the whole chain; term $\Pi'$ accounts for the $h + 1$ locations spanned by the inward-facing side of the remaining $n - h$ positions at each end of the chain. Finally, term $\Pi''$ accounts for the locations covered by the outward facing side of these $n - h$ positions.

If the horizon $h$ is larger than the polymer length $n$, then every $A$-position can interact with every $B$-position on the polymeric scaffold and $q_{max}(n, h) = n^2$. Merging this with [43] and [44] yields

$$q_{max}(n, h) = \begin{cases} n(2h + 1) - h(h + 1), & \text{for } 0 \leq h \leq n - 1 \\ n^2, & \text{for } h \geq n \end{cases}$$ \[45\]

which appears in the main text. The corner cases are covered correctly: $q_{max}(n, 0) = n$ and $q_{max}(n, n - 1) = n^2$. (Note that $h = n$ yields the same result as $h = n - 1$, which is useful below.)

We use [45] to calculate two scenarios. In scenario 1, $h$ is a simple linear function of the length $n$: $h = \xi n$ with $0 \leq \xi \leq 1$. In other words, every $A$ can monitor the same fraction $\xi$ of $B$-binding sites on a polymer of any size. This seems rather unrealistic (and makes $h$ a continuous variable, although that appears to work just fine). However, scenario 1 may serve as a comparison with the subsequent, more realistic scenario 2.
When \( h = \xi n \), \( h \) is always less or equal than \( n \) and the first case of [45] applies. Using \( q_{\text{max}}(n, h) \) with \( h = \xi n \) instead of \( n^2 \) in the first equation of [36] yields

\[
Q_{\text{max}}(\xi) = \sum_{n=1}^{\infty} [n(2h + 1) - h(h + 1)] s^{n-1} = \sum_{n=1}^{\infty} [n(2\xi n + 1) - \xi n(h + n)] s^{n-1} \\
= \frac{1}{\sigma} \left[ (2 - \xi) \sum_{n=1}^{\infty} n^2 s^n + (1 - \xi) \sum_{n=1}^{\infty} n s^n \right] = \xi (2 - \xi) \frac{s(1 + \sigma s)}{(1 - \sigma s)^2} + (1 - \xi) \frac{s}{(1 - \sigma s)^2},
\]

which leads to

\[
Q = p(t_S, t_A, \alpha)p(t_S, t_B, \beta) \left( \xi (2 - \xi) \frac{s(1 + \sigma s)}{(1 - \sigma s)^2} + (1 - \xi) \frac{s}{(1 - \sigma s)^2} \right) \quad [47]
\]

For \( \xi = 1 \), the expression [47] becomes [30], as a horizon that equals the length of any polymer does not affect \( Q_{\text{max}} \). For \( \xi = 0 \) we get

\[
Q = p(t_S, t_A, \alpha)p(t_S, t_B, \beta) \frac{s}{(1 - \sigma s)^2} = p(t_S, t_A, \alpha)p(t_S, t_B, \beta) t_S, \quad [48]
\]

because of \( t_S = s dW/ds \) for the polymer-only system. Thus, for \( \xi = 0 \), we recover the \( Q \) of the simple monovalent scaffold, since in this case the organization of protomers into polymers doesn’t affect catalytic potential. Scenario 1 is shown in Figure S7, panels A and B.

![Figure S7](image)

**Fig. S7.** Interaction horizon scenarios. **A:** \( q_{\text{max}}(n, h) \), equation [45], for scenario 1 when \( h = \xi n \) (0 \( \leq \) \( \xi \) \( \leq \) 1). **B:** \( q_{\text{max}}(n, h) \), equation [45], for scenario 2 when \( h \) is a constant independent of \( n \). The difference to panel A is that the surface of scenario 2, once \( h \) exceeds \( n \), is a quadratic extension of the surface of scenario 1 in panel A at \( \xi = 1 \). **C:** The \( Q \)-surface [47] for scenario 1 as a function of substrate concentration \( t_B \). **D:** The \( Q \)-surface [50] for scenario 2 as a function of substrate concentration \( t_B \). In Figure S8, this surface is compared against the Michaelis-Menten case. The parameter values in C and D are: \( \alpha = \beta = 10^7 \) M and \( \sigma = 10^8 \) M, \( t_A = 15 \cdot 10^{-9} \) M, and \( t_S = 60 \cdot 10^{-9} \) M.

In scenario 2, \( h = \text{const} \) for all lengths \( n \), which means a “hard” horizon independent of polymer size. This scenario is more
realistic. $Q_{\text{max}}(h)$ becomes

$$Q_{\text{max}}(h) = \sum_{n=1}^{\infty} q_{\text{max}}(n, h)\sigma^{n-1}s^n = \sum_{n=1}^{h} n^2\sigma^{n-1}s^n + \sum_{n=h+1}^{\infty} [n(2h+1) - h(h+1)]\sigma^{n-1}s^n$$

$$= \frac{1}{\sigma} \left\{ n^2(\sigma s)^n + (2h+1) \sum_{n=h+1}^{\infty} n(\sigma s)^n - h(h+1) \sum_{n=h+1}^{\infty} (\sigma s)^n \right\}$$

$$= \frac{1}{\sigma} \left\{ \frac{\sigma s(1 + \sigma s) - (\sigma s)^{h+1}(h+1)^2 - (2h^2 + 2h - 1)\sigma s + h^2(\sigma s)^2}{(1 - \sigma s)^3} \right\}$$

$$+ (2h+1)\frac{(\sigma s)^{h+1}(h+1 - h\sigma s)}{(1 - \sigma s)^2} - h(h+1)\frac{(\sigma s)^{h+1}}{1 - \sigma s}$$

$$= \frac{s(1 + \sigma s - 2(\sigma s)^{h+1})}{(1 - \sigma s)^3},$$

yielding

$$Q = p(t_s, t_A, \alpha)p(t_s, t_B, \beta)\frac{s(1 + \sigma s - 2(\sigma s)^{h+1})}{(1 - \sigma s)^3},$$

which is Eq. [7] of the main text. Expression [50] becomes [48] for $h = 0$, as we would expect. As $h$ increases, [50] quickly converges to the infinite horizon case [30], since $\sigma s < 1$ raised to the power of $h$ becomes negligible. Scenario 2 is shown in Figure S7, panels B and D. As suggested in Figure S8, even restrictive structural constraints (small $h$) make only a relatively modest dent in the catalytic potential of the polymerizing scaffold when compared to that of the plain Michaelis-Menten scenario.

7. The discrete case

While we strive for a reasonably self-contained exposition, some details are only asserted for brevity and are developed in a forthcoming manuscript providing a more general treatment of equilibrium assembly.

In the following, we use the same symbols for the binding affinities $\alpha$, $\beta$, and $\sigma$ as in the continuum case, but they must now be understood as “stochastic affinities”. Specifically, if $\gamma'$ is a binding affinity in the continuum case, the stochastic affinity $\gamma$ (in units of molecules$^{-1}$) is related as $\gamma = \gamma'/(AV)$, where $V$ is the effective volume hosting the system and $A$ is Avogadro’s constant. Thus a polymerization affinity of 3 molecules$^{-1}$ in the discrete case corresponds to about $1.8 \cdot 10^{12}$ M$^{-1}$ in a cell volume of $10^{-12}$ L in the continuum setting.
A. Average catalytic potential. Our objective is to calculate the average catalytic potential \( \langle Q \rangle \) of a scaffold mixture, defined as

\[
\langle Q \rangle = \sum_{i=0}^{\min(t_A,n)} \sum_{j=0}^{\min(t_B,n)} ij \langle S_{ij} \rangle ,
\]

where \( S_{ij} \) is any scaffold (polymer or multivalent) with \( n \) \( A \)-binding sites, of which \( i \) are occupied, and \( n \) \( B \)-binding sites, of which \( j \) are occupied. More precisely, \( S_{ij} \) is the set of all configurations, or molecular species, with \( i \) and \( j \) agents of type \( A \) and \( B \) bound, respectively. \( \langle S_{ij} \rangle \) is the average or expected total number of such configurations in an equilibrium system with resource vector \( \hat{t} = (t_A, t_B, t_S)' \in \mathbb{N}_0^3 \). The ' means a transpose. \( t_S \) is typically the number of scaffolds of a given valency \( n \) or the number of protomers in a polymerizing system. When considering mixtures of scaffolds of different valencies \( i \), \( t_S \) is generalized accordingly.)

This raises the need to compute \( \langle S_{ij} \rangle \), which requires a little detour. We start by defining a few well-known quantities.

B. Boltzmann factor of a molecular species. Each molecular species \( Y_i \) has a Boltzmann factor given by

\[
\varepsilon_i = \prod_r \gamma_r,
\]

where \( \gamma_r = \exp(-\Delta G^0_t/\mu_T) \) is the binding constant of the \( r \)-th reaction and the product runs over a series of reactions \( r \) that constitute an assembly path from atomic components (\( A \), \( B \), and \( S \)). Note that, in the discrete case, \( \varepsilon_i \) is not divided by the number of symmetries \( \omega_i \) as in the continuum case (main text leading up to Eq. [1]). The effect of symmetries is accounted for in the state degeneracy, Eq. [54] below, which considers all instances of \( Y_i \) in a given state. As a consequence, \(-kT \log \varepsilon_i\) is not the free energy of formation, but just the internal energy due to bond formation.

C. Boltzmann factor of a state. By extension, the Boltzmann factor of a system state \( \vec{n} = (n_1, n_2, \ldots, n_C)' \), where \( n_i \) is the number of particles of species \( Y_i \), is given by

\[
\varepsilon(\vec{n}) = \prod_{i=1}^C (\varepsilon_i)^{n_i}.
\]

More precisely, [33] is the Boltzmann factor associated with a particular realization of the state \( \vec{n} \), as all atoms are labelled (distinguishable).

D. Degeneracy of a state. A state \( \vec{n} \) is the specification of a multiset of species in which atom labels are ignored. The degeneracy \( d(\vec{t}, \vec{n}) \) of a state \( \vec{n} \) with resource vector \( \vec{t} = (t_1, \ldots, t_T) \) is the number of distinct ways of realizing it by taking into account atom labels. Let \( \mu_{i,j} \) denote the number of atoms of type \( X_j \) contained in one instance of \( Y_i \). For a given resource vector \( \vec{t} \) the set \( \Sigma(\vec{t}) \) of states \( \vec{n} \) that are compatible with it satisfy \( t_j = \sum_{i=1}^C \mu_{i,j} n_i \) for every atom type \( X_j \). Hence, the degeneracy of a state \( \vec{n} \in \Sigma(\vec{t}) \) is given by

\[
d(\vec{t}, \vec{n}) = \frac{\prod_{i=1}^T t_i!}{\prod i! \prod_{i=1}^C (\omega_i)^{n_i}}.
\]

The numerator counts all permutations of the atoms that constitute the system, the first product in the denominator corrects for all orderings among the \( n_i \) copies of species \( Y_i \) and the second product corrects for all symmetries associated with \( Y_i \).

E. The partition function for a given resource vector. As usual,

\[
Z(\vec{t}) = \sum_{\vec{n} \in \Sigma(\vec{t})} d(\vec{t}, \vec{n}) \varepsilon(\vec{n}),
\]

where the sum runs over all admissible states given resource vector \( \vec{t} \). The equilibrium probability of a state \( \vec{n} \) is given by

\[
p(\vec{t}, \vec{n}) = \frac{d(\vec{t}, \vec{n}) \varepsilon(\vec{n})}{Z(\vec{t})}.
\]
The average number of instances of a specific configuration in equilibrium. For a given resource vector \( \vec{t} \) a species \( Y_i \) occurs in various numbers \( n_i \) across the states \( \vec{n} \) in the admissible set \( \Sigma(\vec{t}) \). The average abundance of \( Y_i \), \( \langle n_i \rangle \) then is

\[
\langle n_i \rangle = \sum_{\vec{n} \in \Sigma(\vec{t})} n_i p(\vec{t}, \vec{n}) = \frac{1}{Z(\vec{t})} \sum_{\vec{n} \in \Sigma(\vec{t})} n_i d(\vec{t}, \vec{n}) \varepsilon(\vec{n}).
\]  

[57]

The workhorse for the discrete treatment of the scaffolding systems discussed in the main text is the following Theorem.

**Theorem:**

The average equilibrium abundance \( \langle n_i \rangle \) of species \( Y_i \) in an assembly system with resource vector \( \vec{t} \) is given by

\[
\langle n_i \rangle = g(\vec{t}, Y_i) \varepsilon_i \frac{Z(\vec{t} - \vec{\mu}_i)}{Z(\vec{t})},
\]

[58]

where \( \vec{\mu}_i = (\mu_{i,1}, \ldots, \mu_{i,T})' \) is the atomic content vector of species \( Y_i \); \( g(\vec{t}, Y_i) \) is the number of distinct realizations of a single instance of \( Y_i \) given resources \( \vec{t} \); and \( Z(\vec{t} - \vec{\mu}_i) \) is the partition function of a system in which the atomic resources have been decreased by the amount needed to build one instance of \( Y_i \).

It is immediate from [54] that

\[
g(\vec{t}, Y_i) = d(\vec{t}, Y_i) = \frac{\prod_{j=1}^{T} t_j!}{\prod_{j=1}^{T} (t_j - \mu_{i,j})! \omega_i},
\]  

[59]

where \( Y_i \) denotes a unit vector in the \( Y_i \) direction. We provide a proof of the theorem using generating functions elsewhere. However, to see why the claim holds, we reason as follows. The subset of \( \Sigma(\vec{t}) \) in which we restrict ourselves to states \( \vec{n} \) that contain at least one copy of \( Y_i \) stands in a 1-1 correspondence to the unrestricted state space \( \Sigma(\vec{t} - \vec{\mu}_i) \), because any realization of \( Y_i \) in \( \Sigma(\vec{t}) \) occurs in all possible contexts and these contexts are precisely the states of \( \Sigma(\vec{t} - \vec{\mu}_i) \). The question then is how the degeneracy and the energy content of a state \( \vec{n} \in \Sigma(\vec{t} - \vec{\mu}_i) \) change by adding \( \vec{\mu}_i \) atoms to realize one instance of \( Y_i \). The degeneracy of state \( \vec{n} \in \Sigma(\vec{t} - \vec{\mu}_i) \) is amplified (multiplied) by \( g(\vec{t}, Y_i) \) realizations of \( Y_i \), but one instance of \( Y_i \) is added to those the state already had and so we also need to divide by \( n_i + 1 \) to compensate for indistinguishable permutations within the instances of \( Y_i \), see [54]. Thus, \( d(\vec{t}, \vec{n} + \vec{Y}_i) = (g(\vec{t}, Y_i)/(n_i + 1))d(\vec{t} - \vec{\mu}_i, \vec{n}) \) and the Theorem follows as summarized symbolically:

\[
\frac{1}{Z(\vec{t})} \sum_{\vec{n} \in \Sigma(\vec{t})} n_i d(\vec{t}, \vec{n}) \varepsilon(\vec{n}) = \frac{1}{Z(\vec{t})} \sum_{\vec{n} \in \Sigma(\vec{t} - \vec{\mu}_i)} (n_i + 1) \frac{g(\vec{t}, Y_i)}{n_i + 1} d(\vec{t} - \vec{\mu}_i, \vec{n}) \varepsilon(\vec{n}) = g(\vec{t}, Y_i) \varepsilon_i \frac{Z(\vec{t} - \vec{\mu}_i)}{Z(\vec{t})}.
\]  

[60]

It remains to compute the partition function of the assembly systems discussed in the main text, which is not too difficult and provided in the subsequent section 8.

8. **Partition functions and average catalytic potential**

**A. Polymerizing scaffold without ligands.** Let a state contain \( i \) bonds (not necessarily in the same polymer). Any such state has a Boltzmann factor \( \sigma^i \), where \( \sigma \) is the binding affinity between two scaffold protomers. We count the number of ways to realize \( i \) bonds as follows. Line up the \( ts \) (labelled) protomers and observe that there are \( ts - 1 \) slots between protomers where a bond could be inserted. Thus there are \( \binom{ts - 1}{i} \) ways of inserting \( i \) bonds and the insertion of \( i \) bonds always creates \( ts - i \) molecules. For each choice of \( i \) slots there are \( ts! \) permutations of the protomers. Since the order in which a choice of bond locations creates the \( ts - i \) molecules is irrelevant, we must reduce the label permutations by \( (ts - i)! \) object permutations to obtain the degeneracy \( d_i \) of a state with \( i \) bonds. The partition function is therefore

\[
Z_{ts}^{\text{poly}} = \sum_{i=0}^{ts-1} \sigma^i \binom{ts - 1}{i} \frac{ts!}{(ts - i)!}.
\]  

[61]

The number of possible realizations of a single polymer \( s_n \) of length \( n \) is \( ts!/(ts - n)! \), which yields with [58] for the average number of polymers of length \( n \), \( \langle s_n \rangle \):

\[
\langle s_n \rangle = \frac{ts!}{(ts - n)!} \sigma^{n-1} \frac{Z_{ts-n}^{\text{poly}}}{Z_{ts}^{\text{poly}}}
\]  

[62]

Figure S9 compares the length distributions of equivalent continuum and discrete polymerization systems.
B. Average catalytic potential of the polymerizing scaffold with ligands. Because of binding independence, the partition function of this system is the product of three partition functions: \( Z_{\text{poly}} \times Z_{\text{dimer}} \times Z_{\text{multivalent}} \), with \( Z_{\text{dimer}} \) the partition function of a system in which \( S \)-agents and \( X \)-agents can dimerize with affinity \( \gamma \). \( Z_{\text{multivalent}} \) is simple to obtain: choose \( i \) agents of type \( A \), \( j \) agents of type \( S \), and pair them:

\[
Z_{\text{dimer}} = \min(t_S, t_X) \sum_{i=0}^{\min(t_S, t_X)} \gamma^i \left( \begin{array}{c} t_S \\ i \end{array} \right) \left( \begin{array}{c} t_X \\ i \end{array} \right) i!
\]

Putting this together yields the partition function for resource vector \( \vec{t} = (t_A, t_B, t_S) \)

\[
Z(\vec{t}) = \sum_{k=0}^{t_S-1} \sigma^k \left( \begin{array}{c} t_S \\ k \end{array} \right) \frac{t_S!}{(t_S - k)!} \left[ \sum_{i=0}^{\min(t_S, t_A)} \alpha^i \left( \begin{array}{c} t_A \\ i \end{array} \right) i! \right] \left[ \sum_{j=0}^{\min(t_S, t_B)} \beta^j \left( \begin{array}{c} t_B \\ j \end{array} \right) j! \right]
\]

The total number of realizations, \( g(\vec{t}, \{A_i S_j B_j\}) \) of polymers of length \( l \) with \( i \) \( A \)-agents and \( j \) \( B \)-agents attached, and thus each with Boltzmann factor \( \sigma^{l-1} \alpha^i \beta^j \), is given by

\[
g(\vec{t}, \{A_i S_j B_j\}) = \frac{t_S!}{(t_S - l)!} \left( \begin{array}{c} l \\ i \end{array} \right) \left( \begin{array}{c} l \\ j \end{array} \right) i! j! \]

\[
= \left( \begin{array}{c} l \\ i \end{array} \right) \left( \begin{array}{c} l \\ j \end{array} \right) \frac{t_s!}{(t_s - l)!} \frac{t_A!}{(t_A - i)!} \frac{t_B!}{(t_B - j)!}
\]

where \( \vec{v} = (i, j, l) \) is the composition vector of the configuration and we define for brevity the factorial of a vector as the product of the factorials of its components. Putting all this together yields the average catalytic potential \( \langle Q \rangle \)

\[
\langle Q_{\text{poly}} \rangle = \sum_{l=1}^{t_S} \sum_{i=0}^{\min(l, t_A)} \sum_{j=0}^{\min(l, t_B)} \frac{\sum_{\vec{v}} \# \text{ of interactions} \times \text{total realizations of configurations with } \vec{v}}{\text{average total counts}} \sigma^{l-1} \alpha^i \beta^j \frac{Z(\vec{t} - \vec{v})}{Z(\vec{t})}
\]

C. Average catalytic potential of the multivalent scaffold with ligands. The case of a multivalent scaffold with \( m \) binding sites for \( A \) and \( n \) binding sites for \( B \) follows the lines of section B. For each type of binding sites one can formulate a partition
function in full analogy to \(Z_{\text{poly},t_S,t_X}^{\text{dimer}}\), but with \(m t_S\) (or \(n t_S\)) sites available to bind \(i\) agents of type \(A\) (or \(j\) agents of type \(B\)) to yield a state with Boltzmann factor \(\alpha^i \beta^j\). Thus, the partition function for a multivalent scaffold system is

\[
\begin{align*}
Z_{t_A,t_B,t_S}^{\text{multi}} = \sum_{i=0}^{\min(m t_S,t_A)} \sum_{j=0}^{\min(n t_S,t_B)} \alpha^i \beta^j \binom{t_A}{i} \binom{m t_S}{i} ! \binom{t_B}{j} \binom{n t_S}{j} j! \tag{67}
\end{align*}
\]

The average number of scaffolds loaded with \(i\) ligands of type \(A\) and \(j\) ligands of type \(B\) in a particular configuration then becomes

\[
\langle n_{ij} \rangle = \frac{t_A^i}{(t_A - i)!} \frac{t_B^j}{(t_B - j)!} t_S^\alpha \beta^j \frac{Z_{t_A-i,t_B-j,t_S-1}^{\text{multi}}}{Z_{t_A,t_B,t_S}^{\text{multi}}}. \tag{68}
\]

Finally, for the average catalytic potential we have

\[
\langle Q_{\text{multi}} \rangle = \sum_{i=0}^{\min(t_A,m)} \sum_{j=0}^{\min(t_B,n)} ij \binom{m}{i} \binom{n}{j} \langle n_{ij} \rangle. \tag{69}
\]

D. Remarks on numerical evaluation. While expressions [66] and [69] are explicit, their use with large particle numbers—\(t_S\), \(t_A\) and \(t_B\)—is limited by numerical instabilities (even after efficiency rearrangements). In a separate paper we connect assembly systems with the theory of analytic combinatorics (2), which provides direct approximations based on viewing generating functions as analytic functions over the complex numbers. In our hands, these approximations are not accurate enough over the entire parameter range for the present context. Our figures were therefore generated using the exact expressions [66] and [69], using arbitrary-precision calculations (to 100 significant digits) in \textit{Mathematica} (3), and employing relatively modest particle numbers to keep computation times reasonable.

9. The maximer probability and 1D percolation

The probability of observing the longest possible polymer, given protomer resources, is obtained from [62] by setting \(n = t_S\):

\[
\langle s_{\text{max}} \rangle = \frac{t_S!}{Z_{\text{poly},t_S}^{t_S}} \sigma^{t_S-1}. \tag{70}
\]

This probability is graphed as a function of \(t_S\) and \(\sigma\) in Figure 5A of the main text.

There is an analogy between 1D bond percolation and polymerization at our level of abstraction. The analogy is an exact correspondence in the case of continuum polymerization and bond percolation on an infinite 1D lattice.

A basic quantity in 1D percolation is the mean number of chains (clusters) of size \(n\) normalized per lattice site, which is given by \(p^{n-1}(1-p)^2\), where \(p\) is the probability of a bond between adjacent lattice sites and functions as a parameter. The same expression obtains in terms of the concentration of polymers of length \(n\) normalized per protomer (1, 4):

\[
\frac{s_n}{t_S} = p^{n-1}(1-p)^2. \tag{71}
\]

In the context of polymers, the bond probability is not the primary parameter, but a function of the basic parameters \(t_S\) and \(\sigma\). Following Flory (1), we can express \(p\) as

\[
p = \frac{t_S - W}{t_S} = 1 - \frac{1}{t_S} \frac{s}{1 - \sigma s}, \tag{72}
\]

with \(W\) the concentration of all polymers as defined in [3] for \(a = b = 0\) and given more compactly by [10]. The first equality defines \(p\) in terms of the difference between the maximal possible concentration of objects in the system (\(t_S\)) and the actual concentration of objects; this difference is the concentration of bonds. Using [31] for \(s\) yields

\[
p = 1 - \frac{2}{1 + \sqrt{1 + 4\sigma t_S}}. \tag{73}
\]

Together, expressions [71] and [73] are equivalent to [31] and connect simple polymerization to percolation. As well-known, in the infinite/continuum case, percolation can only occur at \(p = 1\), which is to say in the limit of \(t_S \to \infty\) or \(\sigma \to \infty\).

The analogy persists but the exact correspondence breaks down in the finite, i.e., discrete, case. The percolation probability in the polymerization case is \(\langle s_{\text{max}} \rangle\) as given by [70]. The bond probability, \(p_{\text{bond}}\), is the expected fraction of bonds and can be computed following the arguments that led to [61]. We obtain

\[
p_{\text{bond}} = \frac{1}{t_S - 1} \sum_{i=1}^{t_S-1} i \sigma^i \frac{t_S - 1}{i!} \frac{t_S!}{(t_S - i)!} Z_{\text{poly},t_S}^{t_S}. \tag{74}
\]
In 1D bond percolation, the percolation probability is

$$p_{\text{perc}} = 1 - (1 - p)^2 \sum_{i=0}^{t_S - 2} i p_i^{t_S - 2} = p(t_S - 2) - 1,$$  \[75\]

with $t_S$ the size of the lattice and $p$ the bond probability.

In Figure 5B of the main text we sweep across a range for $t_S$ and $\sigma$. For each $(t_S, \sigma)$ pair we calculate the corresponding $P_{\text{bond}}$ via $[74]$ as the abscissa and $\langle s_{\text{max}} \rangle$ via $[70]$ as the ordinate. This graph is reproduced as Figure S10B for comparison with finite-size bond percolation, Figure S10A. Clearly in $[75]$ $p$ is just a parameter, but in Figure S10A we compute it via $[74]$ using the same sweep over $t_S$ and $\sigma$ as for Figure S10B to make comparison meaningful. The view from percolation is useful because it packages the dependency on $t_S$ and $\sigma$ into the single quantity $p$ (or $P_{\text{bond}}$).

10. Scaling behavior

We refine the notation for the maximer probability $[70]$ to emphasize the dependence on the parameters $t_S$ and $\sigma$,

$$\langle s_{\text{max}} \rangle[t_S, \sigma] \equiv \langle s_{\text{max}} \rangle,$$  \[76\]

in order to note an approximate scaling relation that we observe numerically:

$$\langle s_{\text{max}} \rangle[t_S, \sigma] \approx \langle s_{\text{max}} \rangle[\xi t_S, \xi \sigma],$$  \[77\]

with $\xi > 0$ a dimensionless scale factor. Two systems are approximately equivalent if their protomer numbers and affinities are related by the same scale factor: $t_S^{(1)} = \xi t_S^{(2)}$ and $\sigma^{(1)} = \xi \sigma^{(2)}$. This implies that $t_S^{(1)}/t_S^{(2)} = \sigma^{(1)}/\sigma^{(2)}$ or $r = \sigma^{(1)}/t_S^{(1)} = \sigma^{(2)}/t_S^{(2)}$. 

Fig. S10. Finite size 1D bond percolation and polymerization. A: This panel is panel B of Figure 5 in the main text. It depicts the probability of the maximer $[70]$ as a function of $P_{\text{bond}}$ as given by $[74]$. Each curve represents a particular $t_S$-value for which $\sigma$ sweeps from 1 to 10000 molecules $^{-1}$. $t_S$ ranges from 10 (topmost curve) to 100 (lowest curve) in increments of 10. B: The plot depicts the 1D bond percolation probability $[75]$ as a function of the same bond probabilities used in panel A. The comparison serves to illustrate the difference between 1D bond percolation and polymerization while also emphasizing the analogy. On the other hand, bond percolation on an infinite 1D lattice is equivalent to polymerization described in terms of continuous concentrations.

Fig. S11. Scaling behavior of the maximer distribution. The panels illustrate the approximate scaling behavior of $\langle s_{\text{max}} \rangle$ from different perspectives implied by $[77]$. In all three panels, the ordinate is the maximer probability as given by $[70]$. A: The graph exemplifies the relation $[77]$ by plotting three curves, blue: $\langle s_{\text{max}} \rangle[10, 0.1 \sigma]$, red: $\langle s_{\text{max}} \rangle[100, \sigma]$, and green: $\langle s_{\text{max}} \rangle[1000, 10 \sigma]$ as a function of the affinity $\sigma$. The blue and green graphs are related to the (arbitrary) red baseline graph by scale factors $\xi = 0.1$ and $\xi = 10$, respectively. The red and blue graphs sit on top of each other, while green has a slight (and slightly $\sigma$-dependent) shift to the left. B: This panel illustrates the scaling version $[78]$, comparing red: $\langle s_{\text{max}} \rangle[1000, r 1000]$, and green: $\langle s_{\text{max}} \rangle[10, r 10]$, sweeping along $r$. C: The graph in this panel shows an integer sweep of the scale factor $\xi$, as per $[77]$, for two pairs, $\xi S, \sigma = [10, 5]$ (red), $\xi S, \sigma = [10, 6]$ (blue). The scaling relation is well fulfilled except for very small particle numbers.
The latter says that two systems behave approximately the same if the ratio \( r \) of their respective affinity to protomer number is the same, which yields another way of expressing the scaling observation as

\[
\langle s_{\text{max}} \rangle [t_S^{(1)}, r t_S^{(1)}] \approx \langle s_{\text{max}} \rangle [t_S^{(2)}, r t_S^{(2)}].
\]

These relations are depicted in Figure S11.

11. Unequal ligand concentrations and ligand binding affinities

A. Polymerizing scaffold system. As in Figure 6 of the main text, Figure S12A evidences the \( \sigma \)-dependence of the initial slope in the discrete system and illustrates the effect of ligand imbalance: Once the scarcer ligand, here \( A \), is mostly bound up and the number of scaffold protomers increases further, \( A \)-ligands must spread across an increasingly wider range of length classes, thereby reducing the likelihood of multiple occupancy on the same polymer. As a result, although the binding opportunities for the more abundant ligand, here \( B \), increase (up to the overall prozone peak), \( B \)-particles bound to a particular polymer are less likely to encounter any \( A \) bound to it. The result is a slope reduction compared to a situation in which both ligands are present in equal numbers. A substantive difference between ligand binding constants causes not only a slope reduction prior to the prozone but has, in particular, the effect of delaying the prozone peak considerably beyond what one would expect based on particle numbers alone. It is worth noting that in the Wnt signaling cascade, ligand affinities—enzyme-scaffold, i.e. GSK3β–Axin, and substrate-scaffold, i.e. \( \beta \)-catenin–Axin—are regulated by the signaling process (5, 6).

![Figure S12](image-url)

**Fig. S12.** Effects in discrete and continuum polymerizing scaffold systems. A: The panel illustrates the effects of the polymerization constant \( \sigma \), of ligand imbalance, and of unequal ligand affinities on discrete polymerization. Red, ligand imbalance: \( t_A = 20, t_B = 100, \alpha = \beta = 0.9 \) molecules \(^{-1}\), \( \sigma = 0.01 \) (lower), \( \sigma = 0.1 \) (middle), \( \sigma = 1 \) (upper). Green, unequal ligand affinities: \( t_A = t_B = 20, \alpha = 0.01, \beta = 1 \) molecules \(^{-1}\), \( \sigma = 1 \) molecules \(^{-1}\), \( t_S \) on the abscissa. B: This panel illustrates the effects of ligand imbalance and of unequal ligand binding constants on continuum polymerization. Blue, unequal binding constants: \( \alpha = 10^2 \) M\(^{-1}\), \( \beta = 10^8 \) M\(^{-1}\), \( t_A = t_B = 10^{-7} \) M, \( \sigma = 10^8 \) M\(^{-1}\). Green, ligand imbalance: \( t_A = 10^{-8} \) M, \( t_B = 10^{-4} \) M, \( \alpha = \beta = 10^{-7} \) M\(^{-1}\), \( \sigma = 10^8 \) M\(^{-1}\).

In the continuum case, unlike the discrete case, the initial slope is independent of the polymerization constant \( \sigma \) until a level of protomer abundance is reached sufficient for making polymerization effective, as discussed in section 5 (equation 41). The inflection point at which the slope changes from 1 to 2 (in a log-log plot) will shift accordingly. After that slope change, the responses to ligand imbalance and to differences between ligand binding constants are analogous to the discrete case, as seen in Figure S12B.

Neither ligand imbalance or differences in binding constants appear to affect the downward slope at large \( t_S \) in the continuum or the discrete case.

B. Multivalent scaffold system. The responses to ligand and affinity imbalances in a multivalent scaffold system follow similar lines as in the polymerizing case. When both ligand types are present with the same number of particles, the ligand with higher affinity experiences the prozone later, since the amount of scaffold-bound ligand is higher compared to the other type. This is seen in Figure S13B with the steepening of the downward slope associated with the stronger binding ligand. The situation with ligand imbalance is analogous. The ligand with higher abundance keeps binding while the scarcer ligand is undergoing its prozone; thus the subdued effect on catalytic potential, which, in the example of Figure S13C is mainly holding a constant level until the prozone for the more abundant ligand sets in. Although affinity and number imbalance mimic each other, the affinity imbalance exhibits a much less pronounced plateau around the prozone peak and consequently the drop-off is less sharp than in the case of number imbalance. Extremely high affinity differences would be required to generate a plateau similar to number imbalance. This is seen in the continuum case, shown in Figure S14A, where affinities differ by 7 orders of magnitude. The concentration imbalance in the continuum case yields a similar picture as in the discrete case (Figure S14B).
12. Stochastic simulations

Our analysis of the discrete case focuses on average behavior. Analytic techniques for higher moments are beyond the scope of this contribution and will be presented elsewhere. In lieu of an analytic treatment, we performed several stochastic simulations using the Kappa platform (7, 8) and GNU Parallel (9). Figure S15 displays the essential observations in the context of Figures 3A and 6A of the main text and S11B of this Supplement.

Fluctuations in the binding of ligands translate into Q-fluctuations on the basis of how sites are partitioned into agents. There are three regimes, which we describe in the case of a monovalent scaffold system for simplicity (lowest green curve in Figure S15; green curve in Figure S16; and Figure S17): (i) At low scaffold numbers, prior to the prozone peak, most scaffolds are fully occupied by both ligands. Fluctuations cause transitions between system states with similar Q and variance is therefore low (see red distributions in Figure S17). (ii) Just past the prozone peak, many scaffolds are still occupied by both ligands, but there is an increasing number of singly bound and some empty scaffolds. Unbinding from a fully occupied scaffold is statistically offset by re-binding to the pool of singly-bound scaffolds, which yields a net effect similar to situation (i). However, in addition, singly-bound scaffolds may also lose their ligand. This event is neutral in Q, but free ligands may re-bind an already singly-bound scaffold, thereby increasing Q. Likewise, dissociation from a fully occupied scaffold an re-association with an empty one will decrease Q. As a result of this expanded Q-range, the variance has increased compared to a situation with similar average Q prior to the prozone peak (see green distributions in Figure S17). (iii) Well past the prozone peak, a number of scaffolds are bound by one ligand and many have no ligands at all. Ligand binding fluctuations will mainly shift ligands from singly-bound scaffolds to empty scaffolds with no effect on Q. As a result, Q-variance is now decreasing again (see blue distributions in Figure S17).
Fig. S15. Stochastic simulations. For all stochastic simulations, we used a volume on the order of an human erythrocyte, \( V = 10^{-12} \) L. All summary statistics were computed with 500 samples, each an independent and equilibrated state. A: The solid curves in this panel are identical to those in Figure 3A of the main text. Stochastic simulations were performed by converting deterministic affinities into stochastic affinities as described in the main text (section “The discrete case in equilibrium”) and by converting concentrations into particle numbers at the given volume \( V \). Averages of catalytic potential are indicated by filled squares. Green: polymerizing system at various protomer numbers, descending from top: 36120 molecules (60 nM), 27090 molecules (45 nM), 18060 molecules (30 nM), 9030 molecules (15 nM). Orange: reference Michaelian system with 60200 (100 nM) enzymes. Because of the large numbers of particles, the standard deviation is smaller than the squares at the chosen scale. This panel is meant as a sanity check that simulations at large particle numbers indeed reproduce the continuum picture as we derived it analytically. B: The curves in this panel are identical to those in Figure 6A of the main text and refer to discrete scaffolding systems. Stochastic simulations were performed using the same parameters listed in that Figure. The squares mark the average catalytic potential, which coincides with the theoretical calculations; the error bars mark one standard deviation. In the polymerizing scaffold case, the simulation allowed us to extend the range of the rather time-consuming calculations using the analytical expression66. Note the log-log scale of the axes distorting the error bars, for a linear-log scale see Figure S16. Green: multivalent scaffolds of valencies \( n = 10 \) (upper), \( n = 5 \) (middle), and \( n = 1 \) (lower). Orange: polymerizing scaffold system with polymerization affinities \( \sigma = 10 \) (upper) and \( \sigma = 0.01 \) (lower). Red: polymerizing scaffold system at the same affinity as the lower orange curve, but with twice the number of ligand particles. C: The curves are identical to those in Figure S11B. As in that Figure, \( r \) is the ratio of affinity to the number of protomers. Squares mark the average number of maximers and error bars mark one standard deviation. Green: system with 10 protomers. Red: system with 1000 protomers.

Fig. S16. Variance and noise. A: This panel reproduces a subset of data from Figure S15B on a linear-log scale to enable a more direct visual interpretation of fluctuations. The green curve in this panel corresponds to the lowest green curve in Figure S15B. It belongs to a system of multivalent scaffolds with valence 1. The orange curve belongs to the polymerizing scaffold system and corresponds to the lowest orange curve in Figure S15B. Because the valency of individual scaffolds in both systems is 1, the number of sites on the abscissa corresponds to the number of scaffold agents, polymerizing or not. The main observation is that for the same average catalytic potential (\( \langle Q \rangle \)) the standard deviation is larger after the prozone peak than prior to it. B: This panel recasts the information in panel A by directly displaying the standard deviation (solid curves). The dashed curves (right ordinate) depict the noise, i.e. the ratio of standard deviation to the mean. The main observation here is that the polymerizing system (orange) is significantly less noisy than the monovalent scaffold system (green).
Fig. S17. Distributions of catalytic potential. Panels A (monovalent scaffold system) and B (polymerizing scaffold system) depict the distribution of catalytic potential for a state sampled prior to the prozone peak (10 scaffold particles, red), just past the peak (100 particles, green) and well past the peak (1000 particles, blue). Other parameters as in Figure 6A of the main text.

References
In this chapter, I characterize the binding level of each interaction in the model, before Wnt addition (section 5.2.1) and after Wnt addition (section 5.2.2). Although the vast majority of bonds equilibrate in a matter of seconds, the rest of the mixture does not; the partitioning of bonds is constructing...
large ensembles, which are visualized at the end of each section (section 5.2.1.1 and section 5.2.2.1). Further characterization of the activities of these ensembles is presented in chapter 6.

5.1 Introduction

Wnt signaling is responsible for the establishment of the dorso-ventral axis. It regulates cell proliferation in the intestinal crypt and aids maintenance of bone morphology. It may even be involved in hair loss. Wnt signaling is also frequently misregulated in cancer, with some known driver mutations affecting the pathway directly. Thus, from both a basic research as well as an interventionist perspective, the Wnt pathway is of interest.

5.1.1 Wnt signaling challenges tools, formalisms, and conceptions

Wnt signaling relies on post-translational modification and complex formation. The Wnt ligand is responsible for decreasing the activity of a complex that degrades the β-catenin protein. This complex is referred to in the literature as the “destruction complex”. While β-catenin is found mostly as a structural component of adherens junctions, it can also migrate to the nucleus, where it can promote or repress gene expression, in concert with members of the TCF/LEF1 family of proteins. Hence, the appearance of the Wnt ligand yields changes in gene expression via the inhibition of the destruction complex and consequent accumulation of β-catenin.

The Wnt ligand, in concert with the Fzd and LRP trans-membrane co-receptors, forms a large protein aggregate at the cell surface, the “Wnt signalosome”. This complex antagonizes the destruction complex. In the pre-Wnt state, the destruction complex performs a series of four sequential
phosphorylations on β-catenin, utilizing two distinct kinases, CK1α providing the priming step for subsequent hyper-phosphorylation by GSK3β. These phosphorylations lead to ubiquitination by the SCF-β-TrCP complex. Ubiquitinated β-catenin is then degraded by the 26S proteasome.

The complex formation and post-translational modification seen in Wnt signaling require various considerations.

**Scaffold usage presents a departure from simple enzymatics**  
The kinases of the destruction complex are organized by scaffolding proteins. Neither CK1α nor GSK3β bind β-catenin directly. The phosphorylation events rely on Axin, which acts as a scaffold, bringing the enzymes into proximity of the substrate. This is aided in turn by APC, a scaffold capable of binding Axin and several more β-catenin molecules.

**Protein disorder introduces flexibility**  
Axin contains a disordered region linking the enzyme binding sites to the substrate binding sites. Since the sites within this disordered region are small, binding may not trigger a disorder-to-order transition, conveying some degree of flexibility for the bound ensemble (a “negative noodle”).

**Disordered tails introduce more flexibility**  
In the same vein, the phosphorylation residues on β-catenin are located on a disordered terminal tail. It seems therefore reasonable to posit that, even if Axin were rigid, the phosphorylation sites of a bound β-catenin can nonetheless explore a local volume. Since Axin can polymerize, it seems plausible that an enzyme loaded on one Axin can phosphorylate the substrate bound on one or more neighboring Axins.
Flexibility and scaffolds yield local concentration effects  The notions above suggest a conceptual shift: in the least constrained scenario, the number of substrate binding sites for a given enzyme becomes the length of its Axin polymer plus the contributions from APC. This has consequences for the activity of the destruction complex (chapter 4). A single enzyme bound to an Axin octa-mer might interact with eight substrates, rather than just one. This local concentration effect would be modulated by the binding dynamics of enzymes and substrates to scaffolds, as well the binding dynamics of scaffolds to scaffolds. If any enzyme can act on any substrate that is bound to the same polymer, enzymatic activity will experience a combinatorial effect.

We refer to the number of enzyme-substrate interactions on a scaffold species as its “catalytic potential”. The catalytic potential of a system state is the sum of the catalytic potentials of its species. For a mathematical comparison of the simplest polymeric scaffold system with a Michaelian one, see chapter 4.

Pharmacological ambition requires mechanistic clarity  The above complexes are known to be large, with literature mentions of puncta, fibrils, or signalosomes\textsuperscript{12,74,15,72}. Yet many biochemistry and cell biology textbooks, reviews, and schematics –the tools used to transmit the “state of the art”– often do not mention the formation of larger aggregates (e.g. fig. 5.1), while in contrast biophysical reviews describe them as phase-separated compartments\textsuperscript{17}. Different fields have contrasting descriptions of the mechanisms regulating Wnt signaling. The model does not perform a biophysical simulation, nor does it simulate a whole cell; but it is informed by one and recapitulates a sliver of the other, presenting an avenue for reconciliation. Since Wnt signaling abnormalities are
frequently involved in cancers, the capacity to alter the activity of the destruction complex might be of pharmacological interest. Rather than targeting enzymes, it might be more effective to target the scaffold binding sites to sequester scaffolding scaffolds into non-scaffolding roles.

**Previous modeling approaches abstracted away crucial sources of complexity**

Several mathematical models of Wnt signaling exist, but none address the combinatorial aspects that come with formation of large complexes (e.g. ⁵⁵, ⁵², ⁴⁴, ⁵⁶). To capture the complexity of combinatorial assembly phenomena as they arise in the Wnt system (e.g. Axin’s protein-protein interaction network in fig. 5.2), necessitated a modeling approach that could admit an arbitrary number of possible molecular species.
Figure 5.2: Protein-protein interaction network for AXIN1_HUMAN (UniProtAC: O15169), taken from StringDB on 25 June 2019. Edges represent "molecular action". Minimum interaction score of "highest confidence 0.9": 50 maximum interactions for 1st and 2nd shells.

The choice of modeling framework I used Kappa, a modeling framework where a modified version of Gillespie’s simulation algorithm uses rules to transform a mixture of agents. Kappa can be though of as a programming language in which every instruction is the expression of an empirical mechanism. This framework allowed writing the model incrementally: as interactions were identified in the literature, they were added to the model. Instead of starting with a prior assump-
tion about the principles of Wnt signaling, I started with a Wnt agent, a β-catenin agent, and kept adding interactions and new players within a preset scope of what constitutes canonical Wnt signaling (i.e. it degrades β-catenin, and Wnt addition reduces that degradation). We see building a model as a way of making an inventory of mechanistic knowledge fragments at the specific level of abstraction defined by the Kappa language (22,23,24,25,14). Simulation is a way of discovering what the formalized understanding of these fragments would play as.

5.1.2 The contents of the model

The Wnt model (appendix A) contains the protein-protein interactions of key actors in the canonical Wnt signaling cascade. Proteins known to play a role in Wnt signaling, for which we have mechanistic interactions, are included. Mechanistic refers to interactions that have a clear biophysical meaning at the level of molecular biology: e.g. phosphorylation of [residue] at [site] of [protein] by [enzyme] when [condition satisfied]. Thus, our rules recapitulate protein binding, unbinding, post-translational modification (PTM), and degradation, with the appropriate causal restrictions. E.g. phosphorylation of Threonine 41 on β-catenin by the Glycogen Synthase Kinase 3β (GSK3β) can only occur if the downstream Serine 45 is phosphorylated –the priming site. Ubiquitination of β-catenin on some Lysine* by the Skip-Cullin-F-box β-Transductin Containing Gene complex (SCF-β-TrCP, or just β-TrCP for short) can only occur after Serines 33 and 37 are both phosphorylated –the phosphodegron.

The purpose of the model is to understand what occurs upon addition of Wnt. The model reca-

* Since the ubiquitinated Lysine(s) remain(s) unknown, I use a fictitious site called LysX.
pilulates the assembly of the β-catenin destruction complex, and its antagonist the Wnt signalosome. Since the signaling event occurs on a timescale of minutes, we disregard the much slower transcriptional feedbacks and focus exclusively on the signaling component. We therefore do not include β-catenin’s gene regulatory activities, the interactions with nuclear pore complex, or the regulation of the transcription co-factors by 14-3-3 complexes. Modeling those aspects is beyond our current computational capability and not grounded in sufficiently detailed knowledge. Assembly of the β-TrCP complex and 26S proteasome is also side-stepped by including these as pre-assembled “atoms”.

The set of proteins in the model includes some that are not considered to be elements of canonical Wnt signaling (see fig. 5.3). These mainly concern phosphatases. They were included because there are key events for which they contribute a missing link in the event cascade; I am not arguing they are the one true answer, but rather they are a mechanistically plausible explanation for observed pathway behavior given molecular interactions that are known.

The stochastic binding and unbinding rates in section 5.1.2 yield a stochastic dissociation constant of $\Gamma_D = 100$ molecules. Using a volume of $2.25 \times 10^{-12}$ L, on the order of a mammalian fibroblast, this yields a deterministic dissociation constant of $K_D = 0.1nM$. I recognize that number as extreme, considerably higher than for example the reported dissociation constant of the Axin-APC interaction of $K_D = 50nM$\textsuperscript{55}. This said, the results from section 7.1 using $K_D = 50nM$ and a dimeric APC yield giant components and demixing (fig. 7.4). Moreover, they also show a meta-stable regime when observing the bond frequencies (section 7.1.3.1), which complicates the determination of having reached a steady state. To avoid that uncertainty, and to explore the behavior of the giant components, I use the high affinity values described above to bias their construction.
Figure 5.3: Contact map for the Wnt model. The contact map shows every type of agent that occurs in the model along with its compliment of sites. All bonds that are possible are drawn as black connections between appropriate sites. The set of possible internal states of an agent (e.g. residues and their PTM states) appears as a list in the gray text box of that agent. The contact map reports no causality information, which is contained in the rules listed in appendix A. Initial abundances and parameters presented in section 5.1.2.

The model in this thesis incorporates a substantial amount of information (e.g. fig. 5.2) in an executable form (appendix A) that is more respectful of the biochemical richness than many traditional depictions (e.g. fig. 5.1).

The size of the model. At over 1300 rules, 31 rule families, 18 agent types, 57 binding sites, and 76 state modification sites (e.g. phosphorylated), this is to my knowledge, both the biggest and most detailed model ever made of a metazoan signaling cascade.

This ambition presents challenges:
<table>
<thead>
<tr>
<th>Entity</th>
<th>nM</th>
<th>particles</th>
<th>Entity</th>
<th>nM</th>
<th>particles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axin</td>
<td>100</td>
<td>135,000</td>
<td>PP1</td>
<td>10</td>
<td>13,500</td>
</tr>
<tr>
<td>APC</td>
<td>100</td>
<td>135,000</td>
<td>PP2 sub-unit Aα</td>
<td>10</td>
<td>13,500</td>
</tr>
<tr>
<td>Dvl</td>
<td>100</td>
<td>135,000</td>
<td>PP2 sub-unit Cα</td>
<td>10</td>
<td>13,500</td>
</tr>
<tr>
<td>CK1α</td>
<td>10</td>
<td>13,500</td>
<td>PP2 sub-unit B55a</td>
<td>10</td>
<td>13,500</td>
</tr>
<tr>
<td>GSK3β</td>
<td>10</td>
<td>13,500</td>
<td>PP2 sub-unit B56a</td>
<td>10</td>
<td>13,500</td>
</tr>
<tr>
<td>β-TrCP</td>
<td>10</td>
<td>13,500</td>
<td>CK1δ</td>
<td>10</td>
<td>13,500</td>
</tr>
<tr>
<td>Proteasome</td>
<td>10</td>
<td>13,500</td>
<td>CK1ε</td>
<td>10</td>
<td>13,500</td>
</tr>
<tr>
<td>Fzd</td>
<td>50</td>
<td>67,500</td>
<td>β-catenin</td>
<td>100</td>
<td>135,000</td>
</tr>
<tr>
<td>LRP</td>
<td>50</td>
<td>67,500</td>
<td>Wnt</td>
<td>3</td>
<td>4,050</td>
</tr>
</tbody>
</table>

Parameter                         | Value
---                                | ---
Generic binding rate               | $1 \times 10^{-3}$ molecules$^{-1}\text{ s}^{-1}$
Generic unbinding rate             | $1 \times 10^{-1}$ s$^{-1}$
Generic phosphorylation rate        | 1 s$^{-1}$
Generic dephosphorylation rate      | $1 \times 10^{1}$ s$^{-1}$
Generic trans-membrane receptor dimerization rate | $1.6 \times 10^{-3}$ molecules$^{-1}\text{ s}^{-1}$
Generic trans-membrane receptor dissociation rate | $1.6 \times 10^{-2}$ s$^{-1}$
Generic ubiquitination rate         | $3 \times 10^{-2}$ s$^{-1}$
Generic proteolysis rate            | $2 \times 10^{1}$ s$^{-1}$
Unary factor for intramolecular ring closures | $1 \times 10^{3}$
Entropy factor (dual bond establishment in rigid rings) | 0.1
Influx rate of β-catenin            | $10 \text{ nM s}^{-1}$

Table 5.1: Parameter table. Unless noted otherwise, these are the parameters used throughout the simulations. For interactions where PTM induces affinity changes, depending on how the mechanism is described in the literature, the rules may implement a change to the binding rate, the unbinding rate, or both. These are documented on a case-by-case basis in the rules themselves, appendix A.

**The number of entities** A model where every individual peptide gets a corresponding agent is costly: $500\mu\text{M}$ of β-catenin in a fibroblast-like volume of $\approx 2.25\mu\text{L}$ yields 675,000 proteins, which in Kappa must be represented as individual agents in the reaction mixture. This is in addition to the scaffolds, enzymes, receptors, ligands.
Distinct timescales in the model, the need to explicitly simulate do-undo loops (e.g. via kinase and phosphatase action), as well as the assembly and disassembly of polymers, all contribute to a simulation’s computational cost. Some simulations in this work have taken several weeks to complete.

To understand what happened in a simulation requires the analysis of event races, i.e. the written records of all events that occurred in a simulation. For some runs these trace files can be terabytes in length. Not only are they difficult to store on disk, but their analysis requires access to vast amounts of random-access memory, sometimes more than available in consumer-grade hardware. These analysis can take upwards of a month to run, making mistakes very costly. Doing trace analysis requires dedicated hardware and very careful planning. Analysis of snapshots of the system state need to be sophisticated because of the complexity of the mixture: there was no “off the shelf” software I could access for its analysis, so I had to write it myself (section 3.1).

For these reasons, I present three studies: an overview of the system at 1:1 size in chapter 5, an in-depth characterization of the system down-sized 1:100 in chapter 6, and a further characterization in simplified mixtures and models in chapter 7.

5.1.3 This is not “the one true model” of canonical Wnt signaling

What is actually meant by “canonical Wnt signaling”? Is the canonical Wnt signaling that occurs in intestinal crypt maintenance the same canonical Wnt signaling that occurs in dorso-ventral axis establishment? Is the canonical Wnt signaling that occurs in human intestinal crypts the same as the
canonical Wnt signaling that occurs in nematode intestinal crypts? Is the canonical Wnt signaling that occurs in bone morphology maintenance the same as the canonical Wnt signaling that occurs in hair follicle maintenance?

I believe the answer to the above is no. The expression profile of an adult’s crypt cell is different from the expression profile of a developing embryo: a fraction of the embryo’s Axin appears to be maternally inherited, and so it can be in PTM states unachievable by the embryo itself. Although conserved evolutionarily, the proteins regulating Wnt signaling are not identical across species: Wnt3A is an activator in humans, but a repressor in nematodes. Even within species, cell types have different expression profiles and abundances: not only are proteins expressed at different levels, but they can be spliced alternatively, re-configuring their interactions. For these reasons, it does not seem plausible the signaling cascade operates in identical fashion neither within nor across taxa.

Canonical Wnt signaling is a functional concept, in that mechanistically quite distinct chains of events may be functionally analogous. On that account, it would be enough to say that Wnt signaling is any accumulation of a protein conforming to IPR013284 in response to the presentation of a protein conforming to IPR003817. In the absence of sufficient detail to model any of the above tissue- or species-specific instances, it seems reasonable to settle for a “platonic” version based on a consensus view of interaction fragments conserved across species.
5.2 The assembly of protein aggregates in the Wnt system absent β-catenin

β-catenin is subject to cyclic modifications and de-modifications, mediated by scaffold aggregates. Simulating the assembly of the aggregates along with the modifications of β-catenin at actual cellular abundances is computationally prohibitive. I will report on downsized simulations in chapter 6. In the present chapter, I will focus on the regulation by Wnt of the assembly and composition of the scaffold aggregates in the absence of β-catenin.

5.2.1 Assembly of the destruction state prior to Wnt addition

This section describes the assembly process of the destruction complex, visualized in section 5.2.1.1. The model is initialized with all agents in monomeric form. I simulate the system until the frequency of bonds for the core scaffolds have appeared to reach a steady state.

Axin-APC interactions  Axin proceeds to form polymers, APC proceeds to form dimers, and together they form cross-linked structures. Axin is capable of head-to-tail concatenation of its DIX domain (section A.1.1.2), APC of dimerizing through its Oligo domain (section A.1.1.1), and APC contains three distinct and independent binding sites for Axin (section A.1.1.3). These bonds stabilize quickly, reaching stable levels in a matter of seconds fig. 5.4.

In fig. 5.4, the stochastic bi-molecular dissociation constant is \( \Gamma_D = 100 \) molecules, which in a human fibroblast-like volume of 2.25\( \mu \)L corresponds to a dissociation constant on the order of \( K_D = 0.1nM \). The particle numbers, denoted with pipe symbols as \(| \cdot |\), for Axin and APC are
Figure 5.4: The two scaffold proteins involved in assembly of the destruction complex are Axin and APC. One can explore the stability of the scaffolding component of this complex via the changes in time of the bonds underpinning the interactions of those scaffolds. Axin can polymerize via head-to-tail concatenation of the DIX domain (blue). APC contains three SAMP repeats, each capable of binding separate Axins concurrently (orange, green, red). APC can dimerize through its oligo domain (purple). Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

\[ |Axin| = |APC| = 135,000 \text{molecules, comparable to 100nM}. \] Using eq. (B.1) for a dimerization equilibrium yields an expected bond fraction of 0.97, equivalent to about 132,000 asymmetric dimers or about 66,000 symmetric ones.

This expectation is tracked well by the symmetric APC-APC bonds, as seen in the figure. The average number of Axin-Axin bonds at steady state is roughly 134,800. This number is higher than what the equation predicts, because Axin polymers can undergo ring closure, which proceed with a unimolecular binding rate. The frequency of ring closures is also affected by the cross-linking of Axin polymers by APC. This aspect is not captured by eq. (B.1). Finally, the number of APC-Axin
bonds is determined by the number of SAMP repeats in the mixture, which is thrice the number of APCs. Thus the fraction of SAMP-bound-Axins is expected to be 0.99 of total Axin.

**Occuancy of Axin's $\beta$-catenin binding site**  
Figure 5.5 depicts the state of the $\beta$-catenin binding site on Axin, which can be occupied by either $\beta$-catenin (section A.1.1.8) or by LRP (section A.1.2.5). As there is no $\beta$-catenin in the model, that bond remains at zero. Moreover, the alternate binding partner, LRP, cannot interact with Axin until Wnt is added to the mixture.

![Figure 5.5](image)

**Figure 5.5**: Load of the $\beta$-catenin binding site of Axin, LRP in blue, $\beta$-catenin in orange. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

The rules governing the interactions underlying fig. 5.5 (section A.1.2.5) rely on two key assumptions: (i) until phosphorylated and presumably folded, the C-terminal intracellular “rosette” on LRP is not capable of binding Axin, and (ii) the affinity between LRP and Fzd is zero. I believe both
assumptions are fair given the literature. However, I believe both assumptions merit further empirical investigation. Regarding (i), a disordered domain conceivably exists in a distribution of three-dimensional configurations (92), hence it could spontaneously fold into the rosette configuration and constitute the binding site for Axin. Regarding (ii), both LRP and Fzd are trans-membrane domains, hence the affinity of the two transmembrane segments is likely to be very low, on the order of unspecific interactions, which, however, is not zero. As seen in the biophysical literature, complexes with a plurality of binding sites for other complexes can act as seeds for phase separation (57).

**ENZYMES BOUND TO AXIN** Figure 5.6 shows the amount of enzymes or enzymatic complexes bound to Axin. The scaffold Axin recruits the agents responsible for the full suite of PTMs that lead to \( \beta \)-catenin degradation (sections A.1.1.6, A.1.1.7, A.1.1.13 and A.1.1.14). CK1\( \alpha \) phosphorylates Serine45, which serves as a priming site for GSK3\( \beta \) to phosphorylate Threonine41, which in turn primes GSK3\( \beta \)-phosphorylation of Serine33 and Serine37. Together these two last phosphorylations act as a phosphodegron, which is recognized by the SCF-\( \beta \)-TrCP complex. This complex then ubiquiti-nates an unknown Lysine, serving as a recognition signal for the Proteasome, which finally degrades \( \beta \)-catenin (section A.1.1.11).

As seen in fig. 5.6, the mixture contains Axins bound to both kinases, to the E3 ligase, and to a proteasome. Given abundances of \(|Axin| = 135,000\), \(|CK\alpha| = |GSK| = |bTrCP| = |Proteasome| = 13,500\) and stochastic dissociation constants of \(\Gamma_D = 100\), the expected binding equilibria are consistent with those observed in the plot. The simple dimerization assumption predicts 13,489 dimers, a number recapitulated by the Axin-GSK bond (blue), the Axin-Proteasome bond (green), and the
Figure 5.6: Enzymes and enzymatic complexes loaded on Axin: GSK3β alone in blue, GSK3β and CK1α in orange, β-TrCP complex in red, Proteasome in green. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

Axin-β-TrCP bond (red), all superposed. The ternary complex of CK1-Axin-GSK can be calculated by multiplying the probability of Axin being bound to CK1α (≈ 0.099) with the probability of Axin being bound to GSK3β (≈ 0.099), times the number of Axins in the system, yielding 1,348, consistent with the figure. By a similar argument, the number of Axins bound concurrently to all four enzymatic agents is ≈ 13.

**Occupancy of the β-catenin binding sites on Axin and APC**  The number of free β-catenin binding sites in the mixture does not change, as there is no β-catenin to bind them, fig. 5.7.
Figure 5.7: Load of $\beta$-catenin binding sites on the scaffolds. Ax Axin contains a single binding site for $\beta$-catenin, blue tracks the total abundance of Axin. As APC contains seven such sites, orange tracks seven-fold the abundance of APC. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

Hyperphosphorylation state of the $\beta$-catenin binding sites  The binding sites for $\beta$-catenin on the scaffolds Axin and APC can be phosphorylated (sections A.1.1.12 and A.1.1.15). This modification increases the affinity towards $\beta$-catenin of both scaffolds (sections A.1.1.4 and A.1.1.8). In addition, it enables an additional region of APC to concurrently (as opposed to competitively) bind $\beta$-catenin alongside an Axin. APC can then displace the Axin from $\beta$-catenin (section A.1.1.5). In the absence of Wnt, these bonds appear to reach a steady state in a few seconds fig. 5.8.

The $\beta$-catenin binding site on all Axins reaches full phosphorylation status. However this is not true of APC’s $\beta$-catenin binding sites, the seven features termed “20 amino-acid repeats”, labeled as rpt20AA1 through rpt20AA7. Given non-zero kinase activity and zero phosphatase activity, fig. 5.9,
Figure 5.8: Phosphorylation state of the $\beta$-catenin binding sites. For Axin’s single site, blue. For each of APC’s 20 amino-acid repeats, orange, green, red, purple, brown, pink and gray, superimposed. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

one would expect the phosphorylation level to converge to the maximum value, i.e. the total APC pool.

Figure 5.8 presents a binding process occurring at two timescales, suggesting each of the 20 amino-acid repeats will become phosphorylated on all sites eventually. The first timescale is likely due to the high binding rates used in the model, reflecting the initial recruitment of a large pool of APC into Axin complexes, where the kinases are bound. This is consistent with the phosphorylation rates for the last phospho residue in the largest of the repeats, S1664 in repeat 4 fig. 5.9.

The second timescale warrants further research. APC contains three binding sites for Axins; at equal concentrations, there are three times more SAMP sites than RGS sites (see in fig. 5.4). Let us
Figure 5.9: Rate of phosphorylation (left) and dephosphorylation (right) for Serine 1664 of APC, the last phosphorylated residue of six, on the fourth 20 amino-acid repeat. Phosphorylation of S1664 requires phosphorylation of T1661, itself requiring S1658, itself requiring T1655, itself requiring S1652, itself requiring S1656 (see section A.1.1.15). For the final residue to be phosphorylated, it means the earlier one was, and by recursion, all previous one were (there is no phosphatase action yet). S1664 is therefore indicative of the hyperphosphorylation rate of the fourth repeat. As the fourth repeat contains the most phosphorylation sites among the seven repeats, its dynamics are conceivably the slowest. The stabilization of its dynamics can be assumed to signal the other six repeats have also stabilized, and therefore the (forward) dynamics of the fourth repeat can be representative of the dynamics of all repeats. Nine hundred ninety-nine time points per simulated second. Rate calculated as the ratio of the discrete difference of values divided by the discrete difference of the time axis. Ordinate in seconds, coordinate in molecules per second. Observable patterns defined in appendix C.

consider two configurations of the mixture at steady state, (i) the most permissive and (ii) the most restrictive of APC phosphorylation. In configuration (i) each APC binds a single Axin, and so the entire APC pool can be phosphorylated. In configuration (ii) each APC binds three distinct Axins, and so the $n$ RGS sites of $n$ Axin agents are occupied by the $n$ SAMP repeats present in $\frac{2}{3}$ APCs, leaving two thirds of the APC pool unable to bind an Axin. Configuration (ii) is favored by Axin polymerization and unimolecular bonding. This said, the Oligo site of APC allows it to dimerize, allowing even configuration (ii) to recruit $2 \cdot \frac{2}{3}$ APCs. These interactions and their mechanisms will be further discussed in section 7.1. The results nonetheless suggest there are two pools of APC in the mixture.
We can inspect the final state of that simulation, a snapshot at \( T = 40s \). For combinatorial convenience, the repeat with the smallest number of phosphorylation sites is repeat number five, containing S1857, S1861, and S1863. Visualizing all the agents on the snapshot as a network yields an object difficult to parse, but its nodes can be colored if they match a particular Kappa pattern (see section 3.1). Thus I can visualize where in the mixture are the APCs that are either fully phosphorylated or fully unphosphorylated on repeat 5 fig. 5.10.

Figure 5.10 is not a piece of geometric art, but is the product of the SFDP \(^{(42)}\) algorithm attempting to layout the entire mixture as a plain graph. It shows 0.99% of APC is in either fully phosphorylated or fully unphosphorylated state. Moreover, the force-directed nature of the layout produces a graph where APC appears mainly in three regions, two bands on the graph’s edge, and a sparse one in the center. The inner region contains both phosphorylated and dephosphorylated APC. The inner band of the edge fringe contains both phosphorylated and dephosphorylated APC. The outer band contains only phosphorylated APC. These results would be explained by a fraction of APC strongly connected to the Axin complex (i.e. center region) being phosphorylated, a fraction of APC weakly connected to the Axin complex (i.e. outer fringe) being phosphorylated, and moderately sized APC complexes disconnected from the Axin complex (i.e. inner fringe) not being phosphorylated. Analyzing the snapshot (via KaSaAn section 3.1) for the location of these phosphoforms in regards to the giant component yields:

- \( n_4, 410 \) phosphorylated S1857
- \( n_4, 495 \) phosphorylated S1861
- \( n_4, 506 \) phosphorylated S1863
Figure 5.10: Visualization of entire state at $T = 40s$: coloring all agents (top left), all APCs (top right), APC fully unphosphorylated on repeat 5 (bottom left), and APC fully phosphorylated on repeat 5 (bottom right).

- 114, 378 phosphorylated on all three sites concurrently, of which:
  - 70, 540 present in the largest complex
  - 43, 838 present elsewhere

- 20, 403 for all three sites unphosphorylated, of which:
  - 0 present in the largest complex
• 219 present in neither fully phosphorylated nor fully unphosphorylated form

Of the total APC pool, the bulk is present in either fully phosphorylated or fully unphosphorylated form, with only trace amounts in intermediate phosphorylation states for this repeat, agreeing with the numbers reported in fig. 5.10. Thus the lower than expected number of fully phosphorylated repeats in fig. 5.8 is not due to the observable not including partial APC phosphorylation forms, but rather is due to a large fraction of 20 amino-acid repeats not being phosphorylated at all.

Of the unphosphorylated fraction, it is entirely outside the largest complex, i.e. the destruction complex. Considering there is neither turnover of APC, nor phosphatase activity, affirms the unphosphorylated pool of APC has never been phosphorylated. There is nothing preventing their eventual recruitment into the enzyme-bearing complexes, only site availability. Recalling fig. 5.4, where the oligomerization of APC has stabilized, where all Axins that could bind a SAMP appear to have done it, and where rampant ring and cycle formation re-enforce the connection strength of the complexes, I believe the system has entered a frustrated regime, where rare un-binding events translate to few binding opportunities for unphosphorylated APC to gain a foothold onto the enzyme-bearing complexes.

The mechanisms of APC dimerization, of Axin polymerization, of APC-Axin interaction, and the abundances used, all contribute to a phenomenon whereby the system has generated two comparable pools of APC, a complex-bound one (70, 540 APCs) that is hyperphosphorylated in its entirety, and a diffusing one (64, 241 APCs) that remains in mixed hyperphosphorylated and unphos-
phorylated states for significantly longer. The assembly process of Axin-APC aggregates is further explored, at physiological abundances and affinities, in section 7.1.

**Summary of observations**  The results above suggest the destruction complexes are strongly connected (fig. 5.4), open to β-catenin binding (figs. 5.5 and 5.7) and in a high affinity state (fig. 5.8), and assembled with the necessary enzymatic machinery (fig. 5.6). And although the bonds that structure the core machinery appear to have reached a steady state, there remain processes that have not stabilized, notably where serial PTM is concerned.

**Interactions between Wnt and its co-receptors**  In the absence of Wnt, there are no bonds it can establish with its co-receptors fig. 5.11. Thus the amount of Wnt in the system is zero, as are any amounts of Wnt-containing complexes.

**Interactions of the Wnt co-receptors**  Although there is no Wnt in the system, the co-receptors are capable of homo-dimerizing on their own (section A.1.2.1). These reach a steady state in a matter of seconds fig. 5.12, although much slower than the assembly process observed by Axin and APC (fig. 5.4). Containing 67,500 molecules each of Fzd and LRP, with a stochastic dissociation rate constant of 100, using eq. (B.1), one calculates roughly 32,000 homo-dimers of each.

In fig. 5.12, the observed mean values for the last 10 seconds (one thousand data points) of 29,888 (Fzd, blue) and 29,713 (LRP, orange) are approximating the predicted value, and are likely not yet at steady state. It is worth noting that the LRP-LRP affinity is the same as the Fzd-Fzd affinity, and both receptors are present at the same abundance, and yet the dimerization level of Fzd is slightly,
Figure 5.11: Wnt (blue), Wnt binding to co-receptors LRP (orange) and Fzd (green), and Wnt bound to both into a ternary complex (red). Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

but consistently, higher than the dimerziation of LRP, notably at later time points. The explanation for this asymmetry involves the Dvl scaffold (fig. 5.13), and key assumptions about the operation of the receptors’ cytosolic binding sites (discussed for figs. 5.14 and 5.16).

DVL AND ITS ASSOCIATION WITH FZD Dvl remains S407 unphosphorylated, as the associated kinases are inactive (section A.1.2.2). In spite of this however, an increasing fraction of Dvl is binding the surface receptor Fzd (fig. 5.13), despite the affinity for that interaction being low for S407 unphosphorylated Dvl (section A.1.2.4).

This behavior is a consequence of the way their interaction is interpreted. Rather than assume their affinity is zero (i.e. equivalent to $K_D = \infty$), I assume it is a very small number ($\Gamma_D = 10^6$, 96
Figure 5.12: Wnt co-receptor homo-dimerizations. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

see section A.1.2.4), mediated by a lower binding rate (vs phosphorylated form) rather than by an accelerated unbinding rate. In this case, this assumption yields a time-scale separation, where their binding is very slow, and by extension weak. Given the descriptions in the literature of this interaction, I can not discount this behavior as a model artifact.

The steady state level of the bond will be impacted by the steady state abundance of the proteins. However, considering that protein stability is regulated by signaling cascades, among which Wnt is known to regulate the stability of Axin and possibly Dvl, it is conceivable these proteins exist on a shorter time-scale than their binding equilibria would take to reach. In such a scenario, the observed binding level would stabilize faster and at a lower level than a simple dimerization model would predict, as there would be an additional outflow of Dvl (bound or not) towards the mathematical
Figure 5.13: The binding of Dvl to the membrane-bound Fzd receptor (blue), is regulated by the state of S407, where the unphosphorylated form (green) has lower affinity than the phosphorylated form (orange). Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

empty set via degradation, accelerating the stabilization and decreasing the level of the Dvl pool.

Whether this degradation mechanism can act on monomeric Dvl, or all Dvls, will affect which subpool is affected most. Such hypothesis merit further characterization, notably of protein stability but also of binding/unbinding rates, not just their ratios.

**Dvl co-polymerization with Axin** In the absence of Wnt, Dvl remains Y17 dephosphorylated. The kinases responsible for that phosphorylation remain inactive (section A.1.2.2). The unphosphorylated state disfavors the copolymerization regime with Axin (section A.1.2.3), as seen in fig. 5.14, green, in favor of the homopolymerization of each (section A.1.3.1), as seen in fig. 5.14, red and fig. 5.4, blue. An abundance of 135,000 molecules of Dvl, with a stochastic dissociation rate of
100 predicts a bond abundance of 131, 375, agreeing well with the that observed in fig. 5.14.

Figure 5.14: Phosphorylation state of Y17 (blue, orange), co-polymerization of Axin and Dvl (green), and homopolymerization of Dvl (red). Blue and green and superposed. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

This said, the level of co-polymerization may appear negligible but it is non-zero (fig. 5.15), as the last 10 second average for the Dsh-Axin bonds are \( \approx 110 \) for the head and tail configurations, respectively.

**LRP Hyperphosphorylation** In the absence of Wnt, the level of LRP phosphorylation is zero (fig. 5.16). LRP is phosphorylated on ten sites by GSK3\( \beta \) and CK1. However until those phosphorylations occur, the binding site for Axin is unavailable, preventing its recruitment into the kinase-bound complexes via Axin. The other mechanism of recruitment requires Wnt itself, which simultaneously binds LRP and Fzd.
Summary of observations  These results portray a system with multiple time-scales: certain processes are outright arrested (figs. 5.11 and 5.16), others are proceeding slowly (fig. 5.13), some others are at comparable speeds (figs. 5.8 and 5.12).

Moreover, these results also showcase some of the consequences for the choice of mechanism used in two key interactions: the pre-Wnt interaction of Fzd-Dvl is rare, whereas the pre-Wnt interaction of Axin-LRP is impossible. These reflect two ways of structuring the causality of the interaction:

In Kappa, we refer to two flavors of causality as soft and hard. Soft causality is a consequence of the kinetics of the whole system (e.g. event happens rarely), whereas
hard causality is wired into the mechanism of action (e.g. event requires something).

The rarity of an event is neither an intrinsic property of the rule that fired it nor of its rate, but rather is a consequence of the system having a much higher total activity than the flow through the rule. In contrast, a requirement is very much a property of the rule, i.e. of the mechanism of action. An inhibition done via hard causality arrests a process, whereas done via soft causality merely slows it down.

In the Wnt model, rampant polymerization is able skew the model's behavior in such a way that the rare event increases in frequency to appreciable numbers. This shows how a softly causal inhibition (Fzd-Dvl interaction in fig. 5.13) can be sidestepped in time in a way a hard causal inhibition

\footnote{Under mass-action dynamics, the flow through a rule is the product of the rule's rate times the number of embeddings of its reactant pattern.}
can not (LRP-Axin interaction in fig. 5.5). This said, there are multiple ways of realizing the macro-
molecular description with other mechanisms, concretely to decrease a dissociation equilibrium
\( \Gamma_D \), one could choose to decrease the unbinding rate \( \gamma_{\text{unbind}} \), or increase the binding rate \( \gamma_{\text{bind}} \), or
by some combination of both. Increasing a rate will accelerate the process, therefore the equilibria
downstream of it. Where a system contains additional processes, the choice of mechanism can have
far flung consequences on the rest of the system, at various time-scales. Had I chosen to increase the
unbinding rate of the Fzd-Dvl interaction to reflect lower affinity in phosphorylated state, that bond
would have equilibrated faster, possibly allowing the Dvl fraction to assemble into larger aggregates
independently of any Axin-APC complex. Although worth pursuing, lack of empirical evidence for
such behavior coupled to the very costly time-scale separation that would arise in the model discour-
aged this approach in the present work.

**Activation of PP1, and its activation of CK\( \delta \) and CK\( \epsilon \)** Although present, PP1 remains
inactive, as the requirements for its activation are unmet (section A.1.3.2). As a consequence, it does
not dephosphorylate the auto-inhibitory tails of CK\( \delta \) and CK\( \epsilon \) (section A.1.3.8), which therefore
remain phosphorylated, ergo inactive (fig. 5.17).

**Load of Axin’s CK1 binding site** There are multiple agents that compete to bind Axin on
the same site: PP1 (section A.1.3.7), CK\( \alpha \), CK\( \delta \), and CK\( \epsilon \) (section A.1.1.13). Although numerous
binding partners are present, there remains a large pool of Axin free on that site, fig. 5.18.

Considering enzyme abundances of \( |CK\alpha| = |CK\delta| = |CK\epsilon| = |PP1| = 13,500 \) molecules,
Figure 5.17: Abundance of active of PP1 (blue), of CK1ε with fully unphosphorylated C-terminal tail (orange), and of CK1δ with fully unphosphorylated C-terminal tail (green). CK1δ and CK1ε contain eight phosphorylatable residues in their C-terminus, which can be phosphorylated unimolecularly. When phosphorylated, the tail inhibits the kinase. When fully unphosphorylated, the kinase is maximally active. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

with a stochastic dissociation constant of 100, the dimerization assumption (eq. (B.1)) would predict bond values of 13, 489. For CK1α and CK1δ there is good agreement in fig. 5.18, blue and orange are superposed. Although the numerics agree, that is not a accurate comparison, as there is competition for the binding site. Calculating the dimerization value when considering the full Axin pool as available yields results similar as when considering only a quarter of the Axin sites available:

\[ |Axin| = 135,000 \text{ molecules yields } 13,489 \text{ dimers, yet } |Axin| = 33,750 \text{ molecules yields } 13,434 \text{ dimers.} \]

Indeed, the assumption breaks down clearer when considering the next two binding partners. For CK1ε and PP1, the steady state bond values observed in fig. 5.18 are 5,905 and 8,868 respectively. CK1ε can be recruited by Dvl, while PP1 by CK1δ, CK1ε, and two different subunits of the
Figure 5.18: Load of the PP1/CK1 binding site of Axin. Ten thousand data points per simulated second. Time in seconds. Abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

PP2 assembly. The interaction network is far from simple. Thus the availability and composition of, say the PP2 complexes can change the number of available PP1s, which compete and therefore alter the binding states of the CK1 isoforms.

One can imagine a regulatory mechanism by which high expression of any of these proteins would compete, and eventually displace, the others from Axin. For example, hyperexpression CK1ε would displace CK1α from Axin and by extension from β-catenin. To the extent these proteins are truly competing for the same binding site, their abundance level present regulatory avenues to the pathway. The same would apply to the phosphatases. I have seen scant, if any, research into quantification of phosphatase levels.
Activation of PP2, catalytic and regulatory subunits  In the absence of Wnt, the de-phosphorylation chain that activates PP1 remains inactive (fig. 5.17), and therefore so do the catalytic and regulatory subunits of PP2 (section A.1.3.3), as seen in fig. 5.19.

**Figure 5.19:** Abundance of the active components of the PP2 components, regulatory subunit B55α (blue), regulatory subunit B56α (orange), and the catalytic subunit Cα (green), all three curves superposed. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

Summary of observations  The results in this section present various processes that are fully stopped, awaiting the appearance of Wnt. The assumptions taken for building these interactions are causally hard, by design. In the absence of greater detail for the mechanisms and interactions, I believe hard causality to be a more conservative and simpler framework with which to fill in the blank spaces. Again, this is more a limitation of available experimental detail, rather than the modeling framework.
5.2.1.1 Visualizing the whole state

The abundance of the bonds between the scaffolds is shown in fig. 5.4, but it is profoundly misleading intuition regarding the size of complexes assembled. Exploring the behavior of the entire mixture is complicated by the pleiomorphic nature of these ensembles. One can not define an simple pattern, or variable, or observable, to track their abundance changes; one would need to programatically generate these patterns by traversing the connectivity map, accounting for the resource vector, and generating all possible configurations of all possible assemblies—a combinatorial explosion of molecular species. Kappa allowed me to avoid this explosion in running the model, but the analysis required me to invent new tools to analyze the entire state of the mixture, section 3.1.

The Kappa simulator can write into a text file the entire state of the mixture at any point in time, what we call a snapshot. These files contain all the agents, explicitly stating their connectivity and internal states. For the final state at $T = 40s$, the snapshot is a 83.2MB text file with over four hundred sixty-three thousand lines (text data not shown). As this is a graph, a network representation might seem useful, but as seen in fig. 5.10 such visualizations are difficult to layout with 688, 500 total agents, therefore 688, 500 nodes (in addition to sites, and PTM states, and their labels).

Instead of treating this as a plain graph, I abstract away bond types and internal states, retaining only species size and composition, to produce a patchwork plot of a snapshot*, shown in fig. 5.20.

I define as a destruction complex any complex that contains all four enzymatic activities required for degradation of β-catenin. Assuming monomeric Axins and simple dimerization equilibria for

* See chapter 1 for another example, section 3.1 for an overview on how to read these plots.
Figure 5.20: A snapshot of the pre-Wnt state at $T = 25$, visualized as a patchwork. Concentrations and abundances as stated in section 5.1.2. Each of these three views contains the same number of species (black border boxes), but the area occupied by each species changes: in Size, the area represents the number of agents in the species; in Count, the area represents the number of times that species appears in the mixture; in Mass, the area is the product of the species count times its size. Each species contains boxes (single color, no border rectangles) describing its composition. Snapshots use the same agent-coloring scheme throughout this chapter.

As expected, the network yields a hairball. However, one can color nodes according to their agent
type. With careful programming, a single layout call can be used for multiple rendering passes, allowing one to selectively color nodes on the same network with the same layouts, facilitating the comparison of different renders. Already from fig. 5.21 we observe the giant component composed of two rather distinct globules. In fact, one is enriched in Axin and APC, the other in Dvl and Fzd, fig. 5.22. Recall from fig. 5.13 how Fzd is slowly trickling into the complex. Figure 5.22 confirms this is being mediated by a Dvl mass, tethered to an Axin-APC ensemble.

The distribution of degradation-related agents tracks the location of Axin, specifically the Axin
Figure 5.22: Location of four key scaffolding agents in the giant component seen in fig. 5.21. Snapshots use the same agent-coloring scheme throughout this chapter. Same agent layout as in fig. 5.21.

lobe of the giant component (fig. 5.23). This complex is a destruction complex.

However, the system is not at steady state. Visualizing the evolution in time of the giant complex reveals how Fzd inclusion is slowly but surely yielding a proto-signalosome by presenting bind-
Figure 5.23: Location of four key enzymatic agents in the giant component from fig. 5.21. Snapshots use the same agent-coloring scheme throughout this chapter. Same agent layout as in fig. 5.21.

The inclusion of Dvl into the destruction complex begins as Dvl fibrils incorporate into the Axin punctum. The same two interfaces that allow Dvl head-to-tail concatenation also allow it to bind
Axin’s polymerization interfaces. Thus the Dvl fibrils terminate at Axins. Once bound on one end of the DIX stack, these fibrils are far more likely to bind the other end of their DIX stack, yielding loops of Dvl tethered on both ends to the Axin core. The infiltration of Dvl fibrils into the Axin
punctum remains a very active research area as of this writing.\textsuperscript{73} As for Fzd, the Dvl fibrils present binding sites for Fzd, which slowly starts occupying them. By its capacity to dimerize, Fzd can then act as a branching point, allowing a Dvl fibril with no available DIX sites to recruit more Dvl fibrils. These in turn can bind the Axin core. The rest of the progression follows on these motifs.
5.2.2 Wnt mediated effects

Here I present some of the effects Wnt is orchestrating. I take the model from above, rewind time to $T = 25$ and add Wnt. This allows me to present side by side the state of two worlds, one where Wnt was added (described above), and one where Wnt is orchestrating changes to the mixture. This section is mostly descriptive as challenges of size and computational cost proved prohibitive; however further analysis is done on a downsized model in chapter 6.

Wnt is now present, and so the machinery in the mixture is being remodeled. This means the distinction between a destruction complex and a Wnt signalosome has become fuzzy; one is actively becoming the other. Indeed, empirical evidence argues the post-Wnt state retains some capacity to phosphorylate $\beta$-catenin \(^{(30)}\), suggesting the signalosome itself may retain some capacity to degrade $\beta$-catenin. In this work, if it contains Wnt and is a large ensemble, I will call it a Wnt signalosome. Thus during the signaling event, the destruction complex morphs into a Wnt signalosome, all the while losing some degradatory power.

**Interactions between Wnt and its co-receptors**  As expected, once Wnt appears, its abundance is non-zero. It proceeds to bind both Fzd as well as LRP (section A.1.2.1), yielding the maximal number of fully doubly-bound ligand in tenths of seconds fig. 5.25.

**Interactions of the Wnt co-receptors**  The dimerization steady states of the co-receptor homodimers are increased following Wnt addition (fig. 5.26). The increase for Fzd is markedly higher than that for LRP. From fig. 5.20, the LRP fraction was not a member of any other species,
existing exclusively as either monomers or homodimers. In contrast, Fzd was scaffolding on the Dvl fibrils, yielding a higher local concentration. This explains their pre-Wnt steady state dissimilarity, however marginal.

Post-Wnt, on one hand the Dvl-Fzd interaction is itself increased, highlighting the local concentration effect for Fzd homodimerization. On the other hand for LRP, it will take time for the LRP
fraction of the mixture to be incorporated into the signalosome, because Wnt is limiting (fig. 5.25). This explains how Fzd dimerization can proceed at a faster time-scale than LRP dimerization post-Wnt.

**Dvl and its association with Fzd** Wnt coordinates the S407 phosphorylation of Dvl (sections A.1.2.2 and A.1.3.2), increasing its affinity towards Fzd (section A.1.2.4). There is no process to dephosphorylate Dvl, and so in enough time, the entire pool is expected to converge to phosphorylated state (fig. 5.27).

![Figure 5.27: Color coding as in fig. 5.13. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.](image)

**Dvl co-polymerization with Axin** Wnt also coordinates the phosphorylation of the region associated to Y17 of Dvl (section A.1.2.2). This has the effect of increasing the affinity of the Dvl DIX domain’s interfaces towards Axin’s DIX domain interfaces, while decreasing the affinity toward’s its own (fig. 5.28). In other words, Dvl-Axin interactions become favored (section A.1.2.3), Dvl-Dvl interactions disfavored (section A.1.3.1). This translates to a phenomenon where the Axin and Dvl
polymers become one class of co-polymer, reducing the level of Axin-Axin and Dvl-Dvl interactions in favor of alternating Axin-Dvl and Dvl-Axin interactions (fig. 5.29). The Dvl fibrils and the Axin punctum are dissolving into one another.

**Figure 5.28:** Color coding as in fig. 5.14. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

**Figure 5.29:** Color coding as in fig. 5.15. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

**LRP hyperphosphorylation**  Given Wnt is present, LRP can now dock onto the giant component seen in fig. 5.24, allowing it to come into contact with CK1ε and GSK3β, which hyperphos-
phorylate it (section A.1.2.6). LRP has mechanisms to spontaneously auto-dephosphorylate (section A.1.3.9), but despite these, the entire LRP pool converges to hyperphosphorylated state, with all ten phosphorylation sites phosphorylated.

**Figure 5.30:** Color coding as in fig. 5.16. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

**Summary of observations** Wnt is orchestrating various changes to the mixture. Even with observation windows of 0.2s, some of these changes are already visible.

Based on the intersections of the curves in fig. 5.25, there are three temporal regimes in the early stages of the Wnt signaling event. The first regime is characterized by having comparable amounts of LRP-bound Wnt as there is Fzd-bound Wnt, with low levels of LRP- and Fzd-bound Wnt, suggesting any Wnt is binding either one or the other at similar rates, but rarely both. The similarity is expected as the co-receptors are present in equal concentrations, dimerize (fig. 5.26), and operate with the same affinities. The second regime observes a speedup of the binding to LRP vs. the binding to Fzd, with a concomitant increase in doubly-bound Wnt. The rates of these two processes track each
other, suggesting the increase in LRP-bound Wnt in driven by an increase in doubly-bound Wnt. The fact that the Fzd-bound Wnt fraction does not enjoy this increase suggests the doubly-bound Wnt rise is driven by LRP binding Fzd-bound Wnt, suggesting the assembly was constructed by Wnts who bound Fzd first, then LRP. The third and final regime observes a convergence where all Fzd-bound Wnts are also LRP-bound, but not all LRP-bound Wnts are Fzd-bound. However all three converge to the total abundance of Wnt in the system. Wnt is the least abundant of these three proteins in the model, so it is not surprising it can act as a limiting agent.

Binding to Fzd would allow Wnt to access the Dvl fibrils (fig. 5.27), and via them the Axin complex (fig. 5.29), and the kinases loaded onto it (fig. 5.33). Once an LRP, frequently a dimer, binds a Wnt in such complexes, it would bring two LRPs into kinase exposure, allowing their hyperphosphorylation (fig. 5.30). Once hyperphosphorylated, they can bind Axin directly (fig. 5.32), even if they themselves are not bound to a Wnt.

Thus a Wnt that happens to bind first a Fzd can be brought into contact with an LRP via a large set of unimolecular interactions: via its Fzd’s dimer’s Wnt binding an LRP dimer, via the Dvl network’s Fzd dimers binding a Wnt binding an LRP dimer, and even via the Axin network binding hyperphosphorylated LRP monomers or dimers. In contrast, a Wnt that binds an LRP first has to wait for bimolecular interactions to be brought into contact with a Fzd.

**Axin-APC Interactions**  At these timescales, there is little if any effect on the stability of the Axin-APC interaction. As for the Axin-Axin interaction, it is decreasing (fig. 5.31) at a rate comparable to the increase in the Dvl-Axin interactions (fig. 5.29).
The dissolution of the Dvl fibrils into the Axin punctum is proceeding, though its effects are small by $T = 25.25$. As the Axin-APC interactions remain strong, this argues against a dramatic loss of strength for the graph, as it likely remains crosslinked via APC monomers or dimers stitching together fibrils composed mostly of Axin.

**Load of Axin’s $\beta$-catenin binding site** As LRP is being hyerphosphorylated, its binding site for Axin is becoming available. This translates to an increase for this bond, that would have remained at zero where it not for Wnt’s actions. The bond stabilizes once it saturates the LRP pool, as it is present at lower concentrations than Axin. $\beta$-catenin remains absent in the model, and so any bond it can be part of remains at zero.

**Enzymatic load of Axin** The enzymatic load of Axin is unperturbed. Axin has every capacity to degrade a $\beta$-catenin, except for its own binding site who is slowly being occupied by LRP (fig. 5.32). However, as LRP is present at lower abundances than Axin, it will never occupy all of
the Axin β-catenin binding sites.

Figure 5.32: Color coding as in fig. 5.5. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

Figure 5.33: Color coding as in fig. 5.6. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

Load of the scaffolds’ β-catenin binding sites  The number of unbound β-catenin binding sites in the mixture remains constant, as there is no β-catenin to bind them, fig. 5.14.
**Figure 5.34:** Color coding as in fig. 5.7. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

**Hyperphosphorylation state of the β-catenin binding sites** The activation of phosphatases by Wnt (figs. 5.37 and 5.40) has translated into dephosphorylation of the β-catenin binding sites, both on Axin as well as on APC. In fact, Axin has been almost completely dephosphorylated, where APC retains some marginal members in hyperphosphorylated state.

**Figure 5.35:** Color coding as in fig. 5.8. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

The value reached by the hyperphosphorylated APCs could be explained by a predominant con-
tribution from APCs who divorced from the giant component before they were dephosphorylated, together with a smaller contribution from APCs who achieve hyperphosphorylation in the giant component. The phosphatase would observe maximum catalytic potential as members of the giant component, components of which they are (fig. 5.20), which induces fast turnover (fig. 5.36), and by extension brief duration for any hyperphosphorylated form.

Figure 5.36: Color coding as in fig. 5.9. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Nine hundred ninety-nine time points per simulated second. Rate calculated as the ratio of the discrete difference of values divided by the discrete difference of the time axis. Ordinate in seconds, coordinate in molecules per second.

Moreover, recall from section 5.2.1 that \( \approx 44,000 \) APCs are hyperphosphorylated on repeat 5 and located outside the giant component. It is that pool which could be responsible for preserving
the hyperphosphorylation level of APC in the face of phosphorylation/dephosphorylation rates on the order of $> 2 \times 10^6$, i.e. a couple million events per second.

**Summary of observations** The inhibitory effects of Wnt remain unclear, in large part because in absence of β-catenin, there is no process to inhibit (this is explored in chapter 6). This said, the changes to the composition of the giant component suggest occupation of the β-catenin binding site on Axin by LRP to be a key regulatory event. Not only does this block opportunities for β-catenin PTM reactions, which via the nature of the catalytic potential would observe a combinatorial decrease, but it also acts to scaffold the growth of an even larger complex on top of the already giant component seen previously (section 5.2.1.1). Oligomerization of surface receptors is known to contribute to the growth of signaling complexes in other signaling cascades.

**Activation of PP1, and its activation of CKδ and CKε** By the mechanisms outline in sections A.1.3.2 and A.1.3.3, Wnt is switching the state of PP1 from off to on, a very literal “activation”. This leads to the dephosphorylation of the C-terminal tails of both CKδ and CKε.

The rise and slight decline in CKδ dephosphorylation vs. the steady rise for CKε can be explained by their binding partners in conjunction with the mechanism of phosphorylation vs. dephosphorylation. CKδ sits at the membrane and can recruit Dvl, in addition to Axin, whereas CKε is restricted to Axin only. There are thus more opportunities for the CKδ agents to come into contact with PP1 via the giant component, translating to a higher achievable dephosphorylation rate than that for CKε (fig. 5.38).
Figure 5.37: Color coding as in fig. 5.17. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

Figure 5.38: Phosphorylation rate (right) vs dephosphorylation rate (left) for CK1ε and CK1ε⁺⁺.

On this note, it is worth noting the phosphorylation reaction occurs *intramolecularly*, rather than *unimolecularly*, meaning the auto-phosphorylation reactions do not enjoy a boost to their catalytic potential from utilizing a polymeric scaffold system, unlike PP1. Thus this mechanism means the “activating” dephosphorylation reaction can outperform, by several order of magnitude, the auto-inhibiting phosphorylation reaction; phosphorylation peaks at rates of $\approx 7 \times 10^4$, vs. dephosphorylation reaching rates of $\approx 6 \times 10^6$ fig. 5.38.
Load of Axin’s CK1 binding site  Wnt addition has no visible effect on the occupancy of Axin’s CK1 binding site. The steady state distribution of binding partners observed in the pre-Wnt regime (fig. 5.18) is maintained (fig. 5.39).

![Graph showing time vs. value for Axin binding partners](image-url)

**Figure 5.39:** Color coding as in fig. 5.18. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

Activation of PP2, catalytic and regulatory subunits  Wnt activates PP1 (section A.1.3.2), which proceeds to bind, dephosphorylate, and render active PP2 subunit $B_{55\alpha}$ as well as the catalytic subunit $C_{\alpha}$. Where $B_{55}$ is a component of a PP2 heterotrimeric complex, dephosphorylated $B_{55\alpha}$ allows binding to a subunit $B_{56\alpha}$, which the catalytic $C_{\alpha}$ can dephosphorylate and render active (section A.1.3.3). This miniature cascade allows Wnt to activate PP1, which in turn “activates” a PP2-$B_{55}$ complex, which in turn “activates” a PP2-$B_{56}$ complex. Each of these recognition subunits targets different components of the Wnt signaling system, PP1 mostly Axin (section A.1.3.7), PP2-$B_{55}$ β-catenin (section A.1.3.4), and PP2-$B_{56}$ APC (section A.1.3.4).

Of note, since the assembly of the heterotrimeric PP2 complexes is very well characterized struc-
Figure 5.40: Color coding as in fig. 5.19. Left, effects of Wnt. Right, effects of Wnt overlaid on appropriate time window of Wnt-free system. Ten thousand data points per simulated second. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

...naturally, the ruleset is constructed to reflect the know geometric constraints, section A.1.3.6. The Cx-B55e-A2 trimer is not forming chains, but at most trimers; idem for a B56 assembly. The rigidity constraint is encoded in Kappa by distributing the constraint along the binding sites, establishing constraints for one binding site from another, thus refining the binding events by enumerating the multiple subassemblies: three reversible rules for monomer binding monomer into a dimer, three reversible rules for dimer binding monomer to trimer (section A.1.3.6).

With the mental goal of forming an $A - B - C$ trimer, I encountered two flavors of structuring the ring assembly, flexible vs. rigid. An assumption of geometric rigidity, where the trimer is the largest entity, and chainings like $A - B - C - A$ are impossible, requires distributing the geometric constraint, barring for example an $A - B$ dimer from binding a $C - A$ dimer. As the mechanism is not local to a binding site, this style of “rigid” rings requires writing out molecular species. For a trimer, this yields 6 reversible rules, but combinatorial explosions lurk for larger entities.
In contrast, one can consider the assembly to be “flexible”, restricting the mechanism of binding to a single site, avoiding the combinatorial explosion. This results in larger assemblies, regardless of affinities. Even at infinite affinities, the chainings are impossible to avoid. Moreover, when considering enzymatics, a chaining of say PP2 could end up with two or more catalytic subunits. When contrasting a rigid vs. flexible mechanism for the PP2 heterotrimeric assembly, I observed the chainings, which meant having an increased local concentration, which led to appreciable effects in overall turnover. This choice of mechanism has consequences, specially as one scales to systems with additional scaffold entities.

These ring-like configurations are stable, but also sensitive to assembly problems, like a prozone. An excess of, say structural subunit Aα could sequester Cα into dimers distinct from those reached by the recognition subunits Bξ, thus leading to a mixture dominated by Aα-Cα dimers and Aα-Bξ dimers, which because of the rigidity constraint would be able to interact. Where these interactions can be regulated by PTM, a pool of proteins could be switched into and out of such inhibitory regimes. This underscores the needs to both further characterize the biological mechanisms regulating these interactions, as well as quantify the abundance of phosphatase components.

**Summary of observations** In fractions of a second, Wnt has orchestrated a dephosphorylation cascade that produces three entities capable of dephosphorylating, and thereby inhibiting, the destruction machinery (figs. 5.37 and 5.40). In parallel, it also relieved the autoinhibitory conditions of two kinases (fig. 5.37), also via phosphatase action, which contributing in shifting the affinities.
that actively remodel the destruction complex into a Wnt signalosome (figs. 5.27 and 5.29).

Most of these interactions, their mechanisms, their abundances and affinities, are poorly documented in the literature. But a conservative interpretation, like the one presented here, illustrates the magnitude such effects can achieve. Wnt signaling will remain mysterious until we devote greater time and effort into understanding the role of phosphatases, as they can do so much more than just “inactivate” components.

5.2.2.1 Visualizing the whole state

Wnt is responsible for various changes to the machinery, its assembly and composition. Figure 5.41 showcases the fusion of the two subregions of the destruction complex, the Axin punctum and Dvl fibrils in (fig. 5.24), into a tighter configuration where the Wnt ligand is sitting literally at the center of the emerging assembly. Wnt is concurrently binding the two lobes that characterized the pre-Wnt state, and by extension it strengthens their interaction.

Two salient effects can be seen in fig. 5.41, regarding the Dvl infiltration and the LRP recruitment. The copolymerization of Axin and Dvl is expected to yield an Axin-Dvl mass. The beginings of this effect can be seen in fig. 5.41, as the Dvl mass infiltrates the Axin one, yielding a less defined border between these masses as time progresses. As for LRP, from the contact map (fig. 5.3), it can be seen how Wnt acts as a capstone, bridging the gap so LRP can be recruited into a complex with the priming kinase that phosphorylates it, GSK3β, itself loaded on Axin. Once LRP is primed, and subsequently hyperphosphorylated by CK1δ, it can bind Axin directly, generating a connectivity loop $\text{Wnt:LRP:Axin:Dvl:Fzd:Wnt}$. In fact, this is a degenerate loop, where every component sauf
Figure 5.41: Effect of Wnt addition on the composition, size, constitution of the giant component. Wnt added at $T = 25$. Left panes, all agents. Center panes, Wnt. Right panes, Dvl.

Wnt can dimerize, or full on polymerize. When viewing the nascent signalosome’s composition, and keeping in mind how a force-directed network layout would center nodes based on their connectivity, one can appreciate how Wnt facilitates the fusion of the Axin punctum, the Dvl fibrils, and the
LRP dimers, into a signalosome (fig. 5.42).

Figure 5.42: Location of the five elements scaffolding the ring at the core of the signalosome assembly, snapshot taken at $T = 25.3316s$.

A note about the substantial computational cost in generating figs. 5.41 and 5.42. It took 8.3 million events to reach $T = 25.0s$ from the initial conditions at $T = 0s$; from there on, it took ten
billion events to reach $T = 25,3316s$ – more than two months of non-stop runtime. This is the cost of activating the phosphatases, recruiting them to a giant component where their catalytic potential yields combinatorially large activities (see chapter 4), and having them engage in do-undo loops with their neighboring kinases, themselves also having combinatorially large activities.
5.3 Outlook

The size of the complex The size of the complex should give us pause. On one hand, geometric constraints of protein flexibility would limit the size of the complex. On another hand, in a cell, one would expect other proteins present besides those few dozen included in this model. Through direct competition for binding sites, as well as steric occlusion of adjacent binding sites, those additional proteins are likely to interfere with the assembly of this giant component, thereby limiting its scope. On a third hand, these proteins are likely engaged in additional processes beyond Wnt signaling. For example, in some cells APC has ATP dependent movement along microtubules, concentrating in crystalline deposits at their negative ends; these ends coincide with the adherens junctions (10), where β-catenin is known to accumulate. Is APC “sweeping” the cytosol, gathering β-catenin at these junctions? If so, would it be fair to consider two pools of APC, one engaged with Axin in degradation of β-catenin, the other concentrating it at structural junctions? How much APC is actually available for degradation activity?

This said, both complexes are observed to be large entities. The destruction complex is reported to contain several hundred copies of Axins alone, while the Wnt signalsme is on the order of a ribosome. Some authors describe them as phase-separated complexes, underscoring not only their exisstance, but also their size and complexity. In this regard, the model presented here is more akin to a synthetic environment containing only the proteins outlined, a cell-free or in vitro reconstruction. I believe an empirical version of the model, one with real proteins in a physical test tube, would recapitulate both the assembly as well as the demixing of these complexes. Whether they operate
as prion-like solid assemblies, or liquid-liquid phase separated compartments, or some type of soft matter, I have neither data nor framework to argue in.

The throughput of the machinery A complex that contains several nmol equivalent of each enzyme can boost turnover well past what is achievable via simpler enzyme-substrate interactions, like the traditional Michaelian mechanism (discussed in chapter 4). In this Wnt model, even if there are fewer destruction complexes than expected by nature of the evident demixing behavior, the combinatorial turnover achieved by that one complex would outperform the additive turnover of smaller ones, dramatically more so if one is using monomeric scaffolds as a benchmark. On this premise alone, whatever the size of the β-catenin destruction complex, any polymeric form would possess a higher capacity to degrade substrates than a monomeric version. Understanding the throughput of the destruction complex requires understanding its size and composition, moreso as these have not just linear effects on throughput, but combinatorial ones.

The need to characterize mechanism Our understanding of the machinery orchestrating Wnt signaling forces us to confront the challenge of characterizing the mechanisms of polymerization (e.g. number of binding sites, affinities, phosphostates, hard vs. soft causality restrictions, rigid vs. flexible ring assemblies), the abundance of agents (like Fzd), as well as the mechanisms governing the stability of these proteins. Axin is being degraded, and whether it can be pried from a destruction complex or must be freely diffusion will have repercussions for the size of the Axin puncta, ergo of the rest of the machineries. The same applies to the Dvl family. This model’s prediction of
the dilution of the Dvl fibrils into the Axin punctum absent Wnt is dependent on the time scale of Fzd-Dvl interaction, their mechanism of binding, and the stability of Axin, Dvl, and Fzd itself. Understanding the biological consequences of all these mechanisms on the phenomenology called Wnt signaling, whether for basic science or pharmaceutical research, will require characterizing and quantifying the mechanisms in great molecular detail.

Moreover, human cells also present more than one member of the various protein families, be them Dvl2 vs Dvl3, Axin1 vs Axin2, APC1 vs APC2, not to mention their splicing variants. Characterizing the abundance of Dvl2 is not enough when Dvl3 can copolymerize with Axin and help scaffold Fzds. At hour-long timescales, knowing the initial abundance of Axin1 is not enough when Axin2 is being transcribed as a consequence of Wnt signaling, but may be enough at shorter timescales before Axin2 has had time to be transcribed. The one true model of Wnt signaling will have to incorporate these details, as their mechanisms of action have biological consequences for the signaling cascade.
Activities of large complexes in Wnt signaling

Cell have more than one destruction complex. In chapter 5 I presented behavior where the system was overwhelmingly demixed. Although the complexes described are real, their magni-
tude is an exaggeration for the reasons outlined in section 5.3. This said, fluorescence experiments suggest hundreds of Axin puncta prior to Wnt stimulation.68,72 If we assume their distribution in size and space is uniform, one would reason each assembly has access to some hundreths of the cell’s total abundances of constituents. For this reason, a down-size of the model by a factor of 1 : 100 could be thought of as representing one of these machines. Downsized by that factor, the model presented in this chapter is not meant to be representative of the whole cell but just the behavior of one destruction complex.

6.1 On resizing a stochastic simulation

The resizing mechanism operates by multiplying the assumed volume by a factor $f$. This operation results in multiplication of all initial amounts and division of rule rates, elevated to the rule’s arity minus one. Thus a unimolecular rate is unaffected, a bimolecular rate is divided, and a zeroth order rate is multiplied. This resizing preserves the deterministic rate constants and descriptions, providing a stochastic representation but with increased noise. However, it also makes the system more likely to demix.

Let us consider a linear polymerizing system, as described in chapter 4, specifically figure S11B. When a stochastic system with $t$ total protomers and an affinity of $\sigma$ is resized by a factor $f$, it yields a new system with $ft$ protomers and affinity $\sigma/f$ (when downsizing, $f < 1$).
\[ t_b = t_a \times f \]

and \[ \sigma_b = \sigma_a / f \]

\[ \therefore r_b = \frac{\sigma_b}{t_b} = \frac{\sigma_a / f}{t_a \times f} = \frac{\sigma_a}{t_a \times f^2} = r_a * f^{-2} \]

When downsizing, i.e. \( 0 < f < 1 \) so that \( f^{-2} > 1 \), the ratio grows \( r_b > r_a \). The downsized system will be more prone to demixing than the full-sized one, even though they have the same deterministic description.

A way of explaining this behavior non-functionally is that projecting a deterministic description in a small volume vs. a big volume, i.e. a downsize, yields:

- fewer proteins, e.g.
  - \( [c] = 1.0nM \) in \( V = 1.0L \) yields \( |c| \approx 6.0 \times 10^{14} \) proteins
  - \( [c] = 1.0nM \) in \( V = 1.0pL \) yields \( |c| \approx 6.0 \times 10^{2} \) proteins

- lower stochastic dissociation rate constants, e.g.
  - \( K_D = 1.0\mu M \) in \( V = 1.0L \) yields \( \Gamma_D \approx 6.0 \times 10^{15} \) proteins
  - \( K_D = 1.0\mu M \) in \( V = 1.0pL \) yields \( \Gamma_D \approx 6.0 \) proteins

In other words, the half-saturation point of simple dimerizations has been lowered (i.e. bonds more likely), and the total amount of stuff in the system has also been lowered (i.e. unbound stuff...
less likely). So, while the systems have the same likelihood of, say an icosamer, the full-size system is unlikely to see its maximer (i.e. $10^{14}$mer), whereas the smaller sized system will likely see its own (i.e. $10^3$mer).

For this reason, it should not come as a surprise that the Wnt system that demixed at full size (e.g. chapter 5) will be even more strongly demixed when downsized in this chapter. Therefore, the reader should understand that I am not arguing for or against the existence of the destruction complex or the Wnt signalosome, both of which are experimentally validated. I am merely characterizing the activity of such complexes under particular assumptions about the abundances of their constituents.

6.2 Recapitulation of empirical observations

Empirical observations suggest Wnt induces an $\approx 6$-fold increase in $\beta$-catenin (30), by inhibiting the destruction complex through a reduction in the throughput of the ubiquitination step (38). These results are consistent with the behavior seen in the model.

From section 6.2, we can see the addition of Wnt coincides with an increase in the total level of $\beta$-catenin. This increase is not due to an accumulation of the ubiquitinated form, but instead of the hyperphosphorylated forms. Specifically, all steps upstream of ubiquitination see accumulation of their respective phosphorylated forms, with more S33 & S37, more T41, and more S45. These results are consistent with an inhibition of the ubiquitination step itself. The total amount of $\beta$-catenin increases, but so does the amount of the fully unmodified “naked” form, which is neither
Figure 6.1: Abundance of various forms of $\beta$-catenin through the simulation; Wnt added at $T = 150s$. In reading order from top-left: total amount of $\beta$-catenin, ubiquitinated $\beta$-catenin, S33 and S37 phosphorylated $\beta$-catenin, T41 phosphorylated $\beta$-catenin, S45 phosphorylated $\beta$-catenin, and unmodified $\beta$-catenin. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

phosphorylated nor ubiquitinated. This suggests the entire gamut of pro-degradation activities of the destruction complex has been impaired, with the ubiquitination step particularly so.

The trajectories presented in section 6.2 are representative of the most frequent behavior of the model. An average over these requires exploring the behavior over multiple iterations.

6.3 Multiple steady states accessible to a down-sized model

A downsize is expected to yield higher fluctuations in the model. To explore this behavior, the model was run 30 times, each with a different seed to the random number generator.\footnote{Using GNU Parallel\textsuperscript{84} proved very useful in launching concurrent simulations while balancing CPU load and RAM usage across various physical computers.} This results in 30 independent executions. When visualizing the abundance of $\beta$-catenin, it became clear the
model was not constrained to having one typical behavior section 6.3.

![Figure 6.2: Abundance of β-catenin as a function of time. Thirty independent simulations, one color per simulation (left), and grouped by their β-catenin value at $T = 300$ s (right). Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.](image)

In section 6.3, based on the abundance of β-catenin, it is clear the system can steady itself two regimes, one low and the other high. Some systems appear to lock into a low state early on (e.g. pink), others track the high states for much longer before returning to a low trajectory (e.g. orange), while others commit rapidly to a high state (e.g. red), and yet others marginally slower (e.g. dark blue). These behaviors suggest the presence of two stable attractors in the post-Wnt stable regime. As the difference in β-catenin between these low and high systems is over five-fold, I can classify the systems based on where their levels stabilized. As the low systems are more frequent, I will use them as the base-line. Henceforth, a “low system” refers to a simulation in which the post-Wnt steady state observed an abundance of β-catenin of less than 6,000 agents, whereas a “high” system an abundance over that value.
6.3.0.1 Deconstruct states to identify driving asymmetry

This section is concerned with understanding what drives a system into a high vs. a low state.

Narrowing the observation window  Taking a snapshot of a model that converged to a high state, just before the Wnt addition, and using it as the starting condition to 40 independent simulations, yields again systems converging to one or the other states fig. 6.3.

![Graph of time vs. value for Cat-B_T_o_t_a_l](image)

**Figure 6.3:** Forty independent simulations taking as initial condition the state of the red curve from section 6.3 at $T = 150s$. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

This meandering behavior suggests the choice of final state has not been locked by this time.

Whatever mechanism is driving the systems towards one or the other steady state, must therefore occur after Wnt addition. From section 6.3, average trajectories depart a standard deviation by
$T > 180s$, suggesting the system has locked into an approach to one or the other steady states. Whatever mechanism is driving this, it happens between $T = 150s$ and $T = 180s$.

**Changes in β-catenin abundance resonate on its interaction network**  The changes are not constrained to β-catenin. The interaction of LRP and Axin, and by extension LRP and the kinases, can be modulated by increasing β-catenin levels, section 6.3.0.1; even if the interaction has a low affinity, it is non-zero, and accumulation will eventually start inhibiting other interactions regulated by soft causality (discussed in chapter 5).

![Figure 6.4](image_url)

**Figure 6.4:** Abundance of hyperphosphorylated LRP (all ten sites). Systems grouped by their β-catenin value at $T = 300s$. Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

The reduction in LRP hyperphosphorylation (section 6.3.0.1) will result in reduced opportunities for the Axin-LRP bond, a core interaction underpinning the Wnt signalosome (discussed in section 5.2.2.1). The way the model is constructed, LRP spontaneously dephosphorylates at a low rate (see section A.1.3.9). A decrease in its phosphorylated value means a decrease in its phosphorylation rate, which can be achieved by exclusion from the kinase-bearing components. At later time points, this type of inhibition is conceivable via an accumulation of β-catenin.
**Reduced observed degradation rates**  As the influx of $\beta$-catenin is constant, the increase in its level must be due to a decrease in the observed degradation rate, as seen in section 6.3.0.1.

![Graph 1: Degradation rate of $\beta$-catenin as a function of time. Systems grouped by their $\beta$-catenin value at $T = 300s$. Time in seconds, Value in molecules, rate in molecules per second.](image1)

The proteasome operates at a constant turnover rate, and its abundance is not changing. Therefore a decrease in the observed degradation rate suggests a decrease in the abundance of ubiquitinated $\beta$-catenin.

**Reduced observed ubiquitination rate**  The decrease in the level of ubiquitinated $\beta$-catenin is observed in section 6.3.0.1, together with a decrease in the ubiquitination rate. This is expected as there is no de-ubiquitinating enzyme in the model, and so that species can not be “undone”, just degraded –a process which is impaired (section 6.3.0.1).

**Reduced phosphodegron state**  A decrease in the ubiquitination rate suggests a decrease in the amount of ubiquitinatable $\beta$-catenin, i.e. the amount of phosphodegron, section 6.3.0.1.

The high systems exhibit a lower level of phosphodegron than the low systems section 6.3.0.1.
Moreover, the difference in each of the phosphorylations that make up said phosphodegron are higher than the difference in the phosphodegron levels, suggesting each phosphostate is present, just not on the same β-catenin; i.e. a concurrency issue.

A decrease in the phosphorylation level of S33 and S37 could be attributed to a combination of lower phosphorylation rate and/or higher dephosphorylation rate. However both activities are increased initially for the high systems section 6.3.0.1.

At around $T = 165s$, the high systems so far observe lower degradation activity (section 6.3.0.1), lower phosphodegron abundance (section 6.3.0.1), and lower phosphorylation and dephosphory-
Figure 6.8: Phosphorylations constituting the $\beta$-catenin phosphodegron, S37 (top row), and S33 (bottom row); abundance (left column), observed phosphorylation rate (middle column), and observed dephosphorylation rate (right column). Systems grouped by their $\beta$-catenin value at $T = 300\text{s}$. Time in seconds, Value in molecules, rates in molecules per second.

By $T = 180\text{s}$, $\beta$-catenin has accumulated to such levels (section 6.3) that the observed phosphorylation and dephosphorylation rates rebound, even if the actual levels of each phosphorylated state remain low. As each of S33 and S37 phosphorylated states is low, their combined state is even lower, with the phosphodegron remaining low. These results suggest an upstream reduction of the phosphorylation required for S33 and S37, $T_{41}$ section 6.3.0.1.

Reduced $T_{41}$ phosphorylated state  $\beta$-catenin’s $T_{41}$ must be in phosphorylated state for each of the phosphorylation modifications that constitute the phosphodegron.

As expected, the levels of $T_{41}$ phosphorylation are markedly lower for the systems converging to a high $\beta$-catenin state vs. a low one section 6.3.0.1. This difference becomes perceptible from $T = 163\text{s}$ onward, where the average behavior of the high systems falls outside one standard deviation of the...
behavior of the low systems. However, the observed phosphorylation and dephosphorylation rates for T41 are not significantly different in the high vs. low systems at this time point. It is not until $T = 165s$ that they have clearly separated. This suggests again an upstream impairment, on S45 section 6.3.0.1.

**Reduced S45 phosphorylated state** β-catenin’s S45 site serves as a priming site for T41 phosphorylation. CK1α phosphorylates it, which satisfies the charge distribution requirements for GSK3β to phosphorylate T41.

The level of S45 phosphorylation is markedly lower for the high systems, with a trending differ-
ence as far back as around $T = 157s$ (section 6.3.0.1). This lower lever would result in reduced levels of all components upstream: phosphorylation of T41 requires phosphorylated S45, phosphorylation of S33 and S37 each requires phosphorylated T41, ubiquitination requires both S33 and S37 phosphorylated, and finally degradation requires ubiquitination (see section A.1.1.11). With the first modification at reduced levels, the rest are impaired. However, although S45 phosphorylation levels are lower, both S45 observed phosphorylation and dephosphorylation rates are higher in high systems vs. low systems. In fact, the trend is apparent as far back as $T = 155s$.

Lower levels but higher do-undo activity point to kinase and phosphatase interplay

The behavior described above for S45 suggest a difference in phosphorylation vs. dephosphorylation activities, at the very least on S45, is what is driving convergence to either a high or a low state. The enzymes responsible for these activities are CK1α and the PP2-B55α heterotrimer. The only prerequisites for each is the state they modify: for S45 phosphorylation, CK1α requires an unphosphorylated state; the opposite for the PP2-B55α complex. The turnover rate of each component is constant, as are their abundances. However the phosphatases are regulated indirectly by Wnt; a difference in their activation could contribute to the diverging kinetics, section 6.3.0.1.

From section 6.3.0.1, there is no appreciable difference in the PP1 activation. The difference in overall Cα activation is small. However the distribution of those active Cα subunits shows a consistent difference, with more located in B55α complexes for the high systems than for the low systems. This alone suggests the high systems have a higher capacity to dephosphorylate β-catenin, just by how their active Cα subunits were partitioned among complexes.
**Figure 6.11:** Activation levels for PP1 (top left), the catalytic subunit of PP2 (top right), and for the heterotrimeric PP2 complexes loaded with each of the two recognition subunits in the model, B55a (bottom left) which binds β-catenin, and B56a (bottom, right) which binds APC. Systems grouped by their β-catenin value at $T = 300\, \text{s}$. Time in seconds, Value in molecules, rates in molecules per second.

**Difference in activity would be amplified by a giant component** The difference in active PP2-B55a levels for the high vs. low systems is small, though a consistent trend. These effects would be amplificated by recruitment into the giant Axin-APC complex. From fig. 5.3, one can see there are various mechanisms for recruitment. The simplest being for CK1α onto Axin, section 6.3.0.1.

At a global level, there is no perceivable difference in the recruitment level of CK1α to Axin, section 6.3.0.1. As for the PP2-B55a complex, unfortunately, I lacked the foresight when I launched
these models to define an observable for the PP2-Cζ–PP2-Aζ–PP2-B55a–β-catenin complex. That said, the mechanism of action of PP2-B55a suggests raw bond recruitment to be uninformative; PP2-B56a is the complex that binds APC (see fig. 5.3), but it is PP2-B55a which dephosphorylates β-catenin (see section A.1.3.4), after directly binding it. As β-catenin is actively being degraded, its residency time in a complex would be low, specially compared to Axin. Thus CK1α would be expected to take advantage of the increased local concentration effect of being loaded onto the giant Axin component to an extend the PP2-B55a complex would not. Although APC provides a more stable docking site for a PP2-B56a complex, those Cζ subunits would not dephosphorylate β-catenin, as they require being directly bound to the correct recognition subunit (see sections A.1.3.4 and A.1.3.5).
From chapter 5, we know the system demixes, and at this size more so (section 6.1). Thus analyzing the giant component of the mixture would shed insight into the behavior of the whole, specially when considering the combinatorial nature of the catalytic potential. Although the observable for PP2-B55a is unavailable, I did generate snapshots, which can be analyzed via KaSaAn section 3.1. I can therefore measure the abundance of key agents in the giant component, section 6.3.0.1.

In terms of the scaffolds, the composition of the giant component, i.e. the destruction complex, is not significantly different for the systems converging to a high vs. a low β-catenin state. We can not attribute the diverging kinetics to lower enzymatic recruitment (e.g. decrease of Axin), or lower substrate recruitment (e.g. decrease in APC), section 6.3.0.1, top row.

By \( T = 155 \) the load of the enzyme, CK1α is not appreciably different in the systems converging to a high β-catenin state. In contrast, the recruitment of the substrate β-catenin is trending higher, section 6.3.0.1, middle row. This suggests the impairment of the high systems is not due to a sudden

---

**Figure 6.13**: Load on the giant complex for the various agents presented. Systems grouped by their β-catenin value at \( T = 300 \) s. Time in seconds, Value in molecules, rates in molecules per second.
loss of CK1α. In fact, despite comparable recruitment levels (section 6.3.0.1), the CK1α activity is higher (section 6.3.0.1) in the systems that end up degrading fewer β-catenins (section 6.3.0.1).

The apparent paradox is resolved by looking at the PP2 recruitment, specifically for the Cα catalytic subunit and the B55a recognition subunit section 6.3.0.1, bottom row. As expected, the level of B55a in the giant component tracks at first the APC recruitment, where the bulk of the β-catenin binding sites are located. At later time points, it tracks the β-catenin levels directly. As for Cα, its recruitment is consistently higher throughout the observation window in the systems that converge to a high β-catenin state vs. those converging to a lower one. This suggests the divergence in the systems behavior is due to an increased phosphatase load for the high systems vis-à-vis the kinase load, fig. 6.14.

The systems that would end up converging to a high β-catenin state consistently have a smaller difference in the number of recruited CK1α minus the number of recruited PP2-Cα throughout the observation window, fig. 6.14. This suggests a small, but non-zero difference in recruitment of PP2, coupled with marginally higher levels of active PP2-B55a complexes, just after the moment of Wnt addition, are what drive the divergence.

A plausible narrative The following narrative explains the above behaviors. A system recruits marginally more PP2-B55a complexes to its giant component. This leads to heightened dephosphorylation, together with heightened phosphorylation, of S45, however not enough to yield higher levels of phosphorylated S45 (section 6.3.0.1). Lower levels of phosphorylated S45 result in lower levels of T41 (section 6.3.0.1), itself yielding lower levels of the phosphodegron (section 6.3.0.1),
Figure 6.14: Comparative of the load of the kinases vs. the phosphatase 2 catalytic subunit. Time in seconds. Difference in number of agents.

which is doubly impaired by both having fewer T41 to operate on, as well as actively being partially dephosphorylated (section 6.3.0.1) by PP2-B55a, which is present. Lower levels of phosphodegron amount to lower levels of degradation (section 6.3.0.1), ergo accumulation of β-catenin (section 6.3). As β-catenin accumulates, it will begin occupying an increasing fraction of binding sites, notably on the giant component (section 6.3.0.1). Each bound β-catenin presents an additional binding site for a PP2-B55a complex, which by gaining a foothold into the giant component can further antagonize the kinases and further aid the accumulation of β-catenin, thereby generating a feedback. Eventually,
β-catenin accumulates to the point it inhibits recruitment of PP2-B55α complexes into the Axin complex by prozone inhibition: all the β-catenin binding sites on the Axin complex are occupied by it, all the PP2-B55α complexes are bound to one, but the likelihood that the same β-catenin binds both is decreased. This effect would inhibit the first feedback, yielding a negative feedback to the positive feedback for β-catenin accumulation.

**Effect of poisoning the PP2-Cα** Under this narrative, the high β-catenin state requires PP2 for its maintenance. By taking a snapshot of a system stabilized in a high regime, modifying the internal state of every PP2-Cα into an inert configuration (i.e. “poisoned”), and relaunching the simulation from this perturbed state, I can test this behavior, section 6.3.0.1.

![Figure 6.15](image)

**Figure 6.15**: The red system from section 6.3, taken at \(T = 300\) s, was modified to produce a model with inert PP2-Cα (orange), or allowed to proceed undisturbed (blue). The inert system was “rescued” at \(T = 500\) s, where all copies of PP2-Cα were set to “inactive” state – a state Wnt can activate. Total amount of β-catenin (left) and total amount of “activated” PP2-Cα. Time in seconds, Value in molecules, rates in molecules per second.

A system that had committed to the high state requires active PP2-mediated dephosphorylation to remain at it, section 6.3.0.1. Launching two simulations from the same conditions and using the same seed, but where one has its PP2-Cα set to an inert form, results in the system departing the
high state and converging to the levels observed for a low state (compare with section 6.3). Moreover, modifying that state by rendering its PP2-Cα susceptible to Wnt action (i.e. changing from “poisoned” to “off”), does not result in the system regaining the high state. It is clear PP2-Cα in active form is required for the high state to be maintained, but not sufficient to achieve it. These results agree well with the narrative presented above.

6.3.1 Outlook

The results above prove PP2-Cα’s activity is required for maintenance of the high attractor. The narrative offers a plausible mechanism for the divergence, together with consistent trends that support it. However these are not a proof. The question I would pose is “where it not for PP2’s actions, would the system still achieve the high state?”. Although it may seem so, this question is not definitively answered by just making a perturbation like for the “poisoned” system above. The lack of active phosphatase (or bound, or assembled) would result in certain events not happening, which would result in a different sampling of the random-number-generator sequence. Such differences, as seen in fig. 6.3 are sufficient to switch a system, that would have gone into the high state, into a low state. With such a simple perturbation I would be unable to prove necessity, but just indicate a change in probability. This question requires counter-factual reasoning, for which the Kappa ecosystem has a proof-of-concept tool at kappa-counterfactuals. Further research may go in that area.
6.4 Causal study of the events leading to degradation

As far as a cell’s signaling machinery is concerned, it is sensible to assume it exists at some steady state before the presentation of the Wnt ligand. Likewise, the perturbations engendered by the Wnt addition would converge to some other steady state given enough time. In between these states, is the signaling event, where the mixture is rapidly and dramatically changing, in size (i.e. β-catenin accumulates, section 6.3), in behavior (i.e. destruction activity decreases, section 6.3.0.1), and in configuration (i.e. Wnt signalosomes assemble, section 6.3.0.1). In this section, I characterize the enzymatic events leading to the destruction of β-catenin.

Structure of the analysis Various types of measures have been presented throughout this thesis, from bond abundances (e.g. fig. 6.3), to observed rates of processes (e.g. section 6.3.0.1), from whole-mixture snapshots (e.g. figs. 5.10 and 5.20), to single-complex visualizations (e.g. fig. 5.22) and even to complex composition assays (e.g. section 6.3.0.1). In this section, I present a novel assay, causal analysis over the chain of events that occurred in the simulation, i.e. trace querying (see chapter 25). For the purposes of this section, “event A causes event B” means to say A was the last of its type before B, with B requiring the state of the world produced by A. For example, the last S45 phosphorylation before a T41 phosphorylation is the one responsible for the latter. However, do bare in mind causality is not transitive; if “A causes B” and “B causes C”, it is not fair to say “A causes C”. The last S45 phosphorylation need not be the one that caused degradation.
On the computational cost of this analysis  
Collecting the trace for an entire run, initialization through to signalosome stabilization, yields files on the order of hundreds of gigabytes. Running a query on such files is memory intensive, often requiring over 64 gigabytes of random-access memory, as large portions of the trace require random-access. As for time, the simulation that produces the trace runs in several weeks; querying the trace runs in a few months. For this reasons, I break the simulation into distinct regimes, ergo separate trace files: firstly a burn-in period where the system assembles from initial monomeric and default conditions; secondly a destruction complex regime where the destruction activity has stabilized; thirdly the rest of the simulation, from signaling event to stabilization of the Wnt signalosome. The computation cost of the third period proved to costly for this writing, so I merely present the results for the second period, the pre-Wnt destruction complex regime.

To fit the analysis, the ruleset was modified. Specifically, section A.1.1.11 was altered to read:

1 'KC1A.().bCat(S45)*1' \[\text{bCat(S45[un/ph]/[.])}, \text{CK1a(fake_site[./1])} \Rightarrow \text{Cat-S45-p @ 0. } \{\text{GeneralPhosphorylation}\} \]

2 'KC1A.().bCat(S45)*2' \[\text{bCat(S45[1./]), CK1a(fake_site[1./])} @ \text{inf} \]

3 'GSK.().bCat(T41)*1' \[\text{bCat(T41[un/ph]/[.1]) S45[ph][.]), GSK(fake_site[./1])} \Rightarrow \text{Cat-T41-p @ 0. } \{\text{GeneralPhosphorylation}\} \]

4 'GSK.().bCat(T41)*2' \[\text{bCat(T41[1./]), GSK(fake_site[1./])} @ \text{inf} \]

5 'GSK.().bCat(S37)*1' \[\text{bCat(S37[un/ph]/[.1]) T41[ph][.]), GSK(fake_site[./1])} \Rightarrow \text{Cat-S37-p @ 0. } \{\text{GeneralPhosphorylation}\} \]

6 'GSK.().bCat(S37)*2' \[\text{bCat(S37[1./]), GSK(fake_site[1./])} @ \text{inf} \]

7 'GSK.().bCat(S33)*1' \[\text{bCat(S33[un/ph]/[.1]) T41[ph][.]), GSK(fake_site[./1])} \Rightarrow \text{Cat-S33-p @ 0. } \{\text{GeneralPhosphorylation}\} \]

8 'GSK.().bCat(S33)*2' \[\text{bCat(S33[1./]), GSK(fake_site[1./])} @ \text{inf} \]

9 'CateninB.().bTrCP@1' \[\text{bCat(S33[ph][.1]) S37[ph][.]) LysX[un/ub][.1])}, \text{bTrCP(fake_site[./1])} \Rightarrow \text{Cat-Lys-ub @ 0. } \{\text{GeneralUbiquitination}\} \]
The modification of the event binds the enzyme to the substrate directly through a fake site, and then an infinite-rate rule unbinds them. This re-write allows me to match for that binding event, and thus obtain a match for the enzyme. I use the following query over the trace (see chapter 2 for the formal presentation of this language and technique):

```plaintext
query {
  'deg_u_T', 'Axin-deg_u', 'APC-deg_u', 'size-deg_u',
  → 'id-prot_deg_u',
  'ubi_u_T', 'Axin-ubi_u', 'APC-ubi_u', 'size-ubi_u',
  → 'id-trcp_ubi_u',
  'S33_u_T', 'Axin-S33_u', 'APC-S33_u', 'size-S33_u',
  → 'id-gsk3_S33_u',
  'S37_u_T', 'Axin-S37_u', 'APC-S37_u', 'size-S37_u',
  → 'id-gsk3_S37_u',
  'T41_a_T', 'Axin-T41_a', 'APC-T41_a', 'size-T41_a',
  → 'id-gsk3_T41_a',
  'T41_b_T', 'Axin-T41_b', 'APC-T41_b', 'size-T41_b',
  → 'id-gsk3_T41_b',
  'S45_a_T', 'Axin-S45_a', 'APC-S45_a', 'size-S45_a',
  → 'id-ck1a_S45_a',
  'S45_b_T', 'Axin-S45_b', 'APC-S45_b', 'size-S45_b',
  → 'id-ck1a_S45_b',
  'cre_u_T'
}
```

match deg_u:{ prot_deg_u: Prot(CatDom[1/]), -cat: bCat(LysX[ub][1]) }
and last ubi_u: { trcp_ubi_u: bTrCP(fake_site[.1]), cat:
  → bCat(LysX[un/ub][.1]) } before deg_u
and last S33_u: { gsk3_S33_u: GSK(fake_site[.1]), cat:
  → bCat(S33{un/ph}[.1]) } before ubi_u
and last S37_u: { gsk3_S37_u: GSK(fake_site[.1]), cat:
  → bCat(S37{un/ph}[.1]) } before ubi_u
and last T41_a: { gsk3_T41_a: GSK(fake_site[.1]), cat:
  → bCat(T41{un/ph}[.1]) } before S37_u
and last T41_b: { gsk3_T41_b: GSK(fake_site[.1]), cat:
  → bCat(T41{un/ph}[.1]) } before S33_u
and last S45_a: { ck1a_S45_a: CK1a(fake_site[.1]), cat:
  → bCat(S45{un/ph}[.1]) } before T41_a
and last S45_b: { ck1a_S45_b: CK1a(fake_site[.1]), cat:
  → bCat(S45{un/ph}[.1]) } before T41_b
and last cre_u: { +cat: bCat() } before deg_u

return {
  time[deg_u], count{Axin'}{component[.deg_u]{cat}},
  → count{APC'}{component[.deg_u]{cat}}, size{component[.deg_u]{cat}},
  → agent_id{prot_deg_u},
  time[ubi_u], count{Axin'}{component[.ubi_u]{cat}},
  → count{APC'}{component[.ubi_u]{cat}}, size{component[.ubi_u]{cat}},
  → agent_id{trcp_ubi_u},
  time[S33_u], count{Axin'}{component[.S33_u]{cat}},
  → count{APC'}{component[.S33_u]{cat}}, size{component[.S33_u]{cat}},
  → agent_id{gsk3_S33_u},
  time[S37_u], count{Axin'}{component[.S37_u]{cat}},
  → count{APC'}{component[.S37_u]{cat}}, size{component[.S37_u]{cat}},
  → agent_id{gsk3_S37_u},
  time[T41_a], count{Axin'}{component[.T41_a]{cat}},
  → count{APC'}{component[.T41_a]{cat}}, size{component[.T41_a]{cat}},
  → agent_id{gsk3_T41_a},
}
This single query obtains all the information used in through the rest of the chapter. It matches
every degradation event, labeling it as deg_u. It then matches the last ubiquitination prior to degra-
dation, and labels that as ubi_u. Idem for the phosphodegron serines. However, for T41 there is
an ambiguity; should it match the last T41 phosphorylation prior to S33 phosphorylation, or prior
to S37 phosphorylation? In fact, there is a causality fork, where the same T41 phosphorylation need
not be the same event that lead to S33 and S37 phosphorylation. The same reasoning applies for S45
phosphorylation, where one such could lead to one T41 leading to S33, a different one to another
T41 leading to S37. I therefore match all these causality lineages independently, hence the use of the
_a for the S33 lineage and _b for the S37 lineage, with _u for unique or unambiguous events.

For the destruction complex regime, the lack of phosphatases means each phosphorylation step is
final; there is no enzyme to undo them. This results in the lineages for S33 and S37 having the same
T41 and S45 events. *

* It is worth reminding ourselves that causality is not transitive; the last S45 phosphorylation need not be
the one that lead to degradation, especially when phosphatases are active. While prior to Wnt addition this
mixture does not contain active phosphatases, post-Wnt it does. Harvard Registrar regulations prevent me
6.4.0.1 Do all enzymes contribute equitably?

One can measure the rate of, say GSK3β mediated phosphorylation of β-catenin’s S45 by creating a token in that rule, then calculating the derivative, as shown in section 6.3.0.1. That however has no information how different GSK3β agents contributed to that activity. Perhaps all enzymes contribute the same amount of events, or perhaps a lucky few bind the giant component and are responsible for the bulk of events. Perhaps a software error in the simulator produces skewed behavior.

With the query outlined above, I can obtain the enzyme identifiers, and plot them by their frequency, fig. 6.16.

Based on the data in fig. 6.16, one can not ascribe overwhelming responsibility to any one specific copy of an enzyme, for any of the degradation steps. On average, each enzyme is responsible for, or involved in, \( \approx 97 \) degradations, with the most active contributing to \( \approx 125 \) degradations, the least active to \( \approx 75 \).

6.4.0.2 What is the scaffold composition of these complexes?

The destruction complex is structured by two scaffold proteins, Axin and APC. Axin possess the binding sites for the kinases GSK3β and the CKI family, as well as one binding site for β-catenin. In contrast, APC contains multiple binding sites for β-catenin and none for enzymes (see fig. 5.3).

In considering the catalytic potential of a complex, a heavy imbalance in number of enzyme vs. substrate binding sites can produce lower throughput than a smaller but more evenly composed scaffold.

\[ \text{from making claims about data I can not show, so I will refrain from commenting what preliminary results suggest occurs once Wnt is added to the mixture.} \]
complex. I explore the composition of these complexes, in terms of the number of Axins and APCs in the complex at the degradation step. In fig. 6.17, I present a two dimensional histogram of these events.

In fig. 6.17, we can see the destruction complexes actively destroying $\beta$-catenin show little variability in their Axin content, presenting only three values for their Axin content. The number of APCs in the complex shows greater variability. Post Wnt addition, one would expect these trends to change: Dvl antagonizes APC inclusion, which aids in maintaining the Axin complex (section 7.1), and the Wnt signalosome grows on top of the Axin punctum (see section 5.2.2.1).
6.4.0.3 Do the $\beta$-catenin modifications happen on the same complex?

Once a $\beta$-catenin binds a complex, does the complex change much? Does the $\beta$-catenin unbind, then bind some other complex to continue its modifications? Does ubiquitination happen in the same complex as, say priming, or phosphorylation, or degradation itself? An important challenge in answering this question is the requirement to identify complexes: pleiomorphic ensembles change their configuration, thus technically they become different complexes, but the differences may not be relevant to this question. Let’s consider

From fig. 6.18, once it binds, the suite of modifications occurs in similarly structured complexes. The most changed compositions saw change of at most $\approx 50$ APCs (i.e. $\beta$-catenin number 10303),
Figure 6.18: Composition of the complex, comparing different events. Each complex contains a number of Axin and APC agents. Those serve as ordinate and coordinate in this plot, respectively. An arrow points from the composition of the event responsible, to the event downstream (e.g. last ubiquitination to degradation). The 5 most changed compositions are presented. The change is measured by root-mean-square-deviation of the complex composition, in Axin (top) or APC content (bottom), measured over the causal events leading to β-catenin degradation. From fig. 6.16, as both S33 and S37 causal lineages are the same, there are 6 events per degradation and consequently 6 arrows per color: S45ph → T41ph, T41ph → S33ph, T41ph → S37ph, S33ph → ubiquitination, S37ph → ubiquitination, ubiquitination → degradation.

or two Axins (various). All the post-translational modifications that were responsible for β-catenin degradation happened on the giant component.

6.4.0.4 ARE THE ENZYMES EXCLUSIVELY ON THE GIANT COMPONENT?

Any molecular species loaded with an Axin could conceivable serve as a destruction complex. From fig. 6.17 we know that the bulk of Axin is in the giant component along with APC. Intuition suggests little to none contributions from other complexes. A verification for this would be to query, for every molecular species that could serve as a destruction complex, i.e. every species with an Axin on it, how many β-catenin binding sites did it have, and how many enzymes of the 4 types were loaded on it, CK1α, GSK3β, β-TrCP, and the proteasomes, fig. 6.19. This relied on snapshot analysis, rather than trace querying.
Figure 6.19: Composition of every possible destruction complex, as a function of the number of β-catenin binding sites (one per Axin, 7 per APC), and the number of each respective enzyme type loaded on them.

From fig. 6.19, the bulk of the destruction complexes are in fact the giant component; that said, there exist a non-zero number of putative destruction complexes. These are however small in their enzymatic load, or the number of β-catenin binding sites, and as such their catalytic potential would be combinatorially smaller than that of the giant component. However, post Wnt-addition, one would expect the loss of APC to destabilize the polymeric system.
6.4.0.5 How long does β-catenin wait for each modification step?

From the results above, β-catenin is being primed, hyperphosphorylated, ubiquitinated, and degraded at the mixture’s giant component. Querying the trace, I can obtain the wait time for each final step, for each β-catenin. At over thirteen thousand events, plotting them all as a function of time, unsorted, yields poor results. Instead I construct a stacked bar graph, and average with events close in their degradation time section 6.4.0.5.

The wait time from creation to degradation of β-catenin can be broken down by each of the steps. In the current state pre-Wnt, β-catenin takes a few seconds to be degraded, from the time it was created, with the longest average wait observed at 5 seconds. Of that wait time, the bulk is waiting for S45 phosphorylation. Once primed, hyperphosphorylation occurs rapidly. The next contributor to the overall wait time is the wait for ubiquitination. One ubiquitinated, degradation occurs rapidly.

However, as the causality chain contains a fork, this is not a simple vertical stack: S33 and S37 have independent processes, therefore independent wait times. From fig. 6.16, we know the T41 and S45 lineages are joined, which greatly simplifies the plot: both S33 and S37 wait for the same event, and therefore do not stand on separate bars. Post Wnt however, one would expect this to change as phosphatases become active.
6.4.1 Conclusion

Based on the results above, the giant component is the one destruction complex. Even if there exist Axin and APC containing complexes in the mixture, some sporadically with one or another enzyme (fig. 6.19), those do not contribute any events during the entire simulation (figs. 6.17 and 6.18).
Once β-catenin binds the giant component, it is subject to all enzymatic activities rather quickly section 6.4.0.5. Within the giant component, enzymes contribute comparably to overall kinetics fig. 6.16.
6.5 Discussion

This model is a 1:100 downsize, and as mentioned in the introduction, is more akin to the characterization of one of a cell’s multiple Axin puncta. The results in section 6.3 could be interpreted as a cell having its machinery in various states; a population of destruction complexes could conceivably be distributed between the high and low turnover regimes observed here, with conceivably more diversity in cells as big as a Xenopus egg. Such observations would require sub-cellular assays in single cells, perhaps via super-resolution imaging of the various complexes involved.

As for the results in section 6.4, the destruction complex operates in a highly integrated manner. Having all enzymatic activities required, with ample binding sites for enzymes and substrates, allows for rapid turnover. The regulatory effects Wnt has on the giant component concern overwhelmingly its assembly; from having Dvl infiltrate the Axin punctum, to the ejection of APC, from the blockade of the substrate binding site, to the activation of reversing enzymes already found on the complex, Wnt triggers the accumulation of β-catenin not by modulating enzymes, but by remodeling the very complex that enzymes and substrate require for interaction.

Enzymes do not exist as monolithic entities, but their three dimensional conformations fluctuate in time. As a consequence, their turnover rate is dynamic and can operate in regimes. Were the turnover rates for each enzyme involved in Wnt signaling resolved at the single-molecule level, it would be interest to revisit the results from section 6.4, substituting the single turnover rate with a distribution of rates. Enzymes would contribute more events per second, i.e. turnovers, during their most effective configuration, a circumstance that would be boosted by being docked on a large com-
plex. At a single-enzyme resolution, rapid turnover events induce rapid subsequent events\textsuperscript{61}, and so via an increased catalytic potential, the usage of a scaffold could increase the turnover rate itself.
On varying affinities, abundances, and number of binding sites
causal explanations for them, it also presents behavior than can be extracted and simplified into simpler or smaller models. In this chapter, I present some of these. In section 7.1 I explore the effects of changing the number of Axin binding sites on APC, specifically the consequences for the size of complexes. In section 7.2 I explore the behavior of the model for various parameters, focusing on the sensitivity to changes in the abundance of proteins vs. changes in the affinity of interactions.

7.1 Polymerization of Axin through cross-polymerization with APC can trigger a phase transition into a de-mixed state

7.1.1 Motivation

The polymerization mode of Axin, in a vacuum, yields a geometric distribution of polymer lengths at physically realizable $K_{eq}$ (see main and supplemental chapter 4). However, Axin does not exist in a vacuum. In the typical cell, there exists another scaffold, APC, which contains multiple binding sites for Axin. These presumably allow it to concurrently recruit multiple Axins, thereby allowing the Axin polymers to scaffold on APC, creating a local concentration effect that further grows the polymer. Even in a vacuum, this scaffold scaffolding scaffold can yield a regime where the a large fraction of the mixture assembles into a single complex, i.e. de-mixing.

Recent literature frequently asserts multivalency as required for de-mixing. In our Axin-APC subsystem, the notion of valency would map to the number of binding sites on an APC subcomplex for Axin. APC contains three SAMP repeats, each capable of independently binding an Axin’s RGS domain. Thus a monomer of APC has the capacity to bind three Axins concurrently – a valency of
3. Moreover, APC contains an Oligo domain, through which it can oligomerize, yielding higher valencies. Finally, mutations in APC frequently driving familial adenomatous polyposis, map to truncations that reduce the number of SAMP repeats, among other sites, yielding a valency of less than 3. This natural variability prompted me to explore the impact of valency on the de-mixing behavior.

7.1.2 Methodology

I define a simplified model. This model contains exclusively Axin and APC, at various abundances, with various $K_{eq}$ for the 2 interactions involved: the reversible binding of Axin to another Axin via the two sides of its DIX domain (e.g. a head-to-tail concatenation of the DIX-head and DIX-tail sites), and the reversible binding of Axin to APC via their respective RGS domain and SAMP repeat(s). This simulations are run at a scale of 1:1, and are therefore not subject to the caveat outlined in section 6.1.

In this particular model, I define an APC agent to have a single Axin binding site, and two inactive polymerizing sites. With no rules for polymerization, APC agents will remain in whatever state I initialize them. This allows me to control for the total number of binding sites while changing the valency. For example, the valency of 3 of wild-type monomeric APC is achieved by initializing the mixture with $n/3$ hetero-trimeric APC agents, allowing the mixture to retain $n$ binding sites for Axin; idem for $n/12$ hetero-dodecameric APC agents having $n$ total binding sites for Axin, or $n$ monomeric APC agents. Thus all the systems described here have the same number of APC protomers (and by extension Axin binding sites), but only change in how these are partitioned.
I explore four valency values, together with four affinities. The constant parameters are $[Axin] = 30\, nM$, $[APC] = 30\, nM$, $V = 2.25 \times 10^{-12}L$; varying affinities are realized by keeping a constant stochastic binding rate $\gamma_f$ and varying the unbinding stochastic rate $\gamma_r$.

### 7.1.3 Results

Firstly, I focus on the frequency of the two bonds, Axin to APC and Axin to Axin, in time (section 7.1.3.1) and at steady state (section 7.1.3.2). In parallel, I study the mass distribution between polymeric species in those same systems (section 7.1.3.3). These reveal a sensitivity of mixing to the valency of APC. To illustrate this behavior, I visualize the whole mixture at representative states (section 7.1.3.4).

#### 7.1.3.1 Equilibration times are sensitive to valency

As these are dynamical systems, one must consider the time dimension. Axin is initialized in a fully monomeric state, unbound from APC. The formation of Axin-APC complexes will change in time. One way of tracking their equilibration is to track the number of Axin-Axin and Axin-APC bonds through time. These are presented in section 7.1.3.1.

For low affinities, the bond values cluster together regardless of valency value (e.g. green and red in section 7.1.3.1). However this is not true at medium to high affinities. The former (e.g. orange in section 7.1.3.1) present a meta-stable regime, where the 6-valent trace converges to the bond value observed for lower valencies, but at $T \approx 0.9$ departs and more than doubles in value, growing to track the 12-valent system. Observing the 12 valent trajectory, this behavior is also present at $T \approx 0.3$.  

173
Figure 7.1: Changes in time of the Axin-Axin bond (top) and for Axin-APC (bottom), for all parameters assayed. Plots colored by either shared valency (left) or affinity (right). Time in seconds, value in molecules, $K_D$ in nM.

At high affinities, all systems converge to their own values, at least within the observation window.

The bond frequency may not be as informative for the overall state of polymerization, as discussed in section 5.2.1.1. The presence of a meta-stable regime in section 7.1.3.1 suggests additional processes happening beyond the mere frequency of bonds changing. A different metric is to measure the size of the largest assembly, fig. 7.2, as those are the entities that require the most bonds,
therefore the most time, to assemble.

\[ K_D = 1000 \text{nM}, \text{valency of 1} \]

\[ K_D = 100 \text{nM}, \text{valency of 1} \]

\[ K_D = 50 \text{nM}, \text{valency of 6} \]

\[ K_D = 10 \text{nM}, \text{valency of 1} \]

Figure 7.2: Size of the largest complex at various time points for the systems described above. From section 7.1.3.1, for each affinity, the slowest convergence to steady state occurs for lower affinities, hence most systems present their slowest system for a valency of 1. For \( K_D = 50 \text{nM} \), the systems are clearly not at steady state; the valency of 6 in particular exhibits the most visible changes, hence it is presented here. Time in seconds, number of agents in molecules.

Based on section 7.1.3.1 and fig. 7.2, it is fair to assume some of the systems have equilibrated, but not all. As expected, the lower the affinity, the higher the unbinding rate, the faster they equilibrate. However, even though all systems that share a \( K_D \) will have the same binding and unbinding rate, their times to equilibration are not comparable. The higher the valency, the faster they equilibrate. This effect is a reflection of the unimolecular binding potentials, which by an argument of increased
local concentration, operate at a faster binding rate that bimolecular interactions.

7.1.3.2 Bond frequency, at constant affinity, is sensitive to valency

For the systems that appear equilibrated, we can measure the average bond content. As the concentrations of Axin and APC are equal, the frequency fraction will be the number of bound APC sites divided by the total number of APC binding sites. Given the structure of the model, this is the number of bound APC agents divided by their total. A similar argument applies for the Axin-Axin bond. For the final time-points of the simulation, this statistic is presented in fig. 7.3.

Intuitively, the higher the affinity, the more frequent the bond should be. Figure 7.3 tracks this well, with higher valencies presenting more frequent bonds at same affinity values. The magnitude of this effect decreases with decreasing affinity. For high to medium affinities, increasing the valency increases the bond likelihood. At $K_D = 50\text{nM}$, the difference between a monomeric APC and a dimeric APC translates to at least a 2-fold bond likelihood increase (orange vs. green in fig. 7.3). The effect for a tetrameric APC is close to 3-fold.

7.1.3.3 Mass distribution is sensitive to valency

Although informative, the frequency of a bond is not necessarily fruitful insight revealing the overall state of the mixture. Assuming the system is Ergodic, we can visualize the mass distribution of a state, and for those systems equilibrated, average over states taken at regular intervals. The mass distributions are presented in fig. 7.4.
Figure 7.3: Bond frequency of the Axin-APC interaction for various affinities, under varying assumptions of APC valency. The higher valencies for $K_D = 50\text{nM}$ are shown for completeness; as those are not at steady state, their standard deviation is expected to be higher.

When visualizing the polymer mass distribution, increasing the valency from one to three has a dramatic effect in the number of monomers, the smallest species, frequently halving their abundance. At all affinities, further increasing the valency shifts mass into larger complexes, for some this leads to eventually populating the heptacontakilamer (70,000) region of polymer sizes, though this

* When visualizing polymer distributions, I find the mass to a good metric for identifying where the bulk of the system is. Visualizing raw abundances (i.e. number of $n$-mer as a function of $n$), is difficult as the coordinate axis has to convey the abundance of both the giant component (between zero and one) as well as that of the constituting monomers (between zero and several million).
effect is attenuated at weaker affinities.

7.1.3.4  Size of machinery is sensitive to valency

From the results above, it appears at low affinities bonds are infrequent (fig. 7.3), whereas at high affinities the system frequently is forced to saturation (section 7.1.3.1), yielding concomitant demixing (fig. 7.4). To visualize the repercussions of valency for the size of the Axin-APC complexes, I focus on the middle affinity $K_D = 50nM$, and visualize the mixture in fig. 7.5.

**Figure 7.4:** Mass distribution in polymeric species for various levels of affinity, under varying assumptions of APC valency. A species with 10 Axins and 2 APCs would have a mass of 12 per copy of that species. States were sampled from the same time interval as fig. 7.3.
Figure 7.5: Patchwork diagrams of equilibrated mixtures for various valencies of the Axin-APC interaction. Mass, size, and count are measured in molecules. Each mono-colored rectangle represents an agent type, red for APC and blue for Axin; each black bound rectangle represents a species, composed often of different kinds of agents, ergo differently colored sub-rectangles.

In fig. 7.5a, a valency of 1 (e.g. mutated APC) yields a mixture dominated by monomers: the most abundant species is present 28,471 times, the mass is present mostly in a species class whose mass is 28,471, and the species size is maxed at 13 (decatri-mer) while showing overall similar size distribu-
tions throughout. From the mass view, the bulk of the mixture is in the form of a pure-APC species. From the size view, the only pure-APC species is near the top-right, indicative of the smallest sizes. As APC has a valency of 1, it can’t form dimers, so the smallest species is a monomer. This snapshot is dominated by APC monomers.

In fig. 7.5b, a valency of 3 (i.e. monomeric APC) yields again a fragmented mixture, but overall richer, with more kinds of molecular species (i.e. there are more boxes than for valency 1). It is still dominated by the monomers: the largest species is a tetradeca-mer, and the size distribution is overall flat, suggesting a monomer fraction is contributing the observed maximum mass contribution by being present at that same count level of $14,198$. It is worth pointing out that in this view, a monomeric APC with valency 3 would appear as a trimer of agents $APC$, having a mass of three. This effect essentially rules-out any contribution of an $APC$ monomer, suggesting the monomer fraction present is Axin’s.

In fig. 7.5c, a valency of 6 (i.e. dimeric APC) yields a peculiar state, where a giant component of size $54,811$ has captured $\approx 67\%$ of the mixture into a single entity. Giant component not withstanding, the distribution of complex sizes is quite flat, and the most frequent species are homo-Axin entities. However, the difference between a monomeric APC and a dimeric APC results in the system demixing at this affinity.

In fig. 7.5d, a valency of 12 (e.g. tetrameric APC) yields a demixed state, with $\approx 92\%$ the mixture being recruited to a single entity of size $75,269$. The most abundant species remain pure-Axin entities, likely monomers and small polymers, but all of these amount to a small fraction of the mixture.

Visualizing the largest complex of these respective states (fig. 7.6) confirms the observations out-
Figure 7.6: Largest complex found at the final observed timepoint, $T = 3.1$, for the varying systems at $K_D = 50$ outlined in fig. 7.5, same order. Valencies clockwise from top-left: 1, 3, 6, 12.

Whether APC operates a monomer (fig. 7.5b), as a dimer (fig. 7.5c), or as a higher-order polymer (fig. 7.5c) can have profound consequences for the size of the destruction complex, fig. 7.6. Mutations that reduce the number of Axin binding sites can, one hand drive a decrease in its size, and on another reduce the richness of species (fig. 7.5a). These effects coupled would reduce the overall
capacity for β-catenin destruction. As a corollary, interventions that introduce scaffolding potential for Axin (e.g. synthetic peptides with multiple SAMP repeats) could rescue loss-of-SAMP mutations.

7.1.4 Discussion

From section 7.1.3.1, specifically the existence of meta-stable regimes, raises the importance of characterizing not only the equilibrium rate constants $K_{eq}$ of a process, but also the forward and reverse rates that govern it. Biological process could be abstracted as modulating the binding rate (e.g. render accessible a terminal tail), the unbinding (e.g. phosphorylation inserts a repulsive charge), or even both. As these alter the response time of the processes they are involved in, they will speed up or slow down subcomponents of a signaling cascade; from memory-effects to race conditions, these will have consequences difficult to predict in complex systems. Moreover, during a signaling event, the system may not have time to reach a steady state, but may exist only in a transient, or meta-stable regime.

“If the only thing you understand is the steady state, but the system never reaches steady state, you understand nothing”

—a Harvard professor teaching the Systems Biology 200 course

As for the demixing behavior, it is clear from sections 7.1.3.3 and 7.1.3.4 this behavior is very sensitive to the distribution of binding sites. The literature on demixing, on phase-separated liquid-liquid transitions, on giant components, specially when applied to biological systems, frequently cites requirements for “degeneracy”, for “multivalency”, for “disorder”\textsuperscript{63,91,5}. Kappa and by exten-
sion this model has no information on protein flexibility, size, geometric constraints, and so can’t recapitulate notions like order, thereby assuming full disorder. As for degeneracy and multivalency, these are represented formally by the number of binding sites. The work here illustrates that the combination of multiple binding opportunities, with little to no geometric impediment, facilitates assembly. Where the scaffolds themselves can act to scaffold other scaffolds, this assembly leads to a runaway effect, where the scaffolding facilitates further growth of the scaffold complex. In a biological context, this raises the importance of characterizing and quantifying the structure of proteins and complexes, not just as static entities, but as dynamic distributions of shape. If phosphorylation of an Oligo domain allows it to go from dimerization to polymerization (e.g. by activating an additional dimerization interface in addition to the one known), the destruction complex could double in size, which would combinatorially boost its catalytic potential. Compared to protein production, phosphorylation is a fast change. Such behavior would allow a cell to rapidly adapt to changes. Likewise, such regulatory inroads into Wnt signaling suggest novel ways of mis-regulating the pathway.
7.2 Size and composition of the destruction complex exhibits sensitivity to protein levels and robustness to affinity strength

7.2.1 Motivation

Axin polymerizes by head-to-tail concatenation. This mechanism yields polymers of varying lengths, as determined by the affinity and abundance of the protein. The literature already speaks of the “fibrils”, “puncta” and other “complexes” containing Axin. These would be expected to be sensitive to the affinities and concentrations of the entities that make up the core of these complexes, specifically the abundances of the scaffolds Axin and APC, and the affinities describing Axin polymerization, APC dimerization, and Axin-APC recruitment. Most of these parameters are poorly reported in the literature. For the Axin concentration various numbers are reported, 0.02nM for Xenopus egg extract, 30nM for Human intestinal epithelial cell lines, or 150nM for Human epithelial kidney lines. For volumes of $V_{\text{Xen}} = 1.010^{-6}$L (BNID: 106735), $V_{\text{Int}} = 2.010^{-12}$L, and $V_{\text{Kid}} = 1.8$L, these yield discrete protein abundances of $|Axin|_{\text{Xen}} = 1.210^7$ molecules, $|Axin|_{\text{Int}} = 3.610^4$ molecules, and $|Axin|_{\text{Kid}} = 1.610^5$ molecules.

7.2.2 Methodology

I use the ruleset from chapter 5, downsize by 1:100, and vary two parameters: the abundance of the Axin agent, and the interaction affinity of the Axin-Axin, Axin-APC and APC-APC interactions. This model contains $\beta$-catenin, flowing in at a constant rate of 10nMs$^{-1}$, yielding a stochastic and downscaled rate of 135molecules$^{-1}$.
7.2.3 Results

I explore two aspects characterizing the activity of the destruction complex at steady state. In section 7.2.3.1 I present the changes in time of key bonds that scaffold the destruction complex, confirming the various processes have reached steady state. In section 7.2.3.2, I explore the changes in the size of complexes caused by various levels of scaffold concentration. In section 7.2.3.3, I characterize the activity of these destruction complexes, both in terms of their overall degradation capacity, as well as the partial assemblies for the four enzymatic agents involved in β-catenin degradation.

7.2.3.1 Are the complexes at steady state?

As the rest of the analysis hinges on averaging over states in time, the complex distributions must have converged for their average behavior to be meaningful. I therefore present the changes in time for key processes, in fig. 7.7 the bonds that assembly the degradation machinery, in fig. 7.8 the changes in size of the largest assembly, and in fig. 7.9 the changes in time the rate of degradation.

In fig. 7.7, the abundance of Axin is far more predictive of the bond value equilibration than the interaction affinity, echoing results from section 7.1. However, from section 7.1.3.1 one might recall the stabilization of bond abundance is not necessarily informative of the stabilization of complex size; the size of the largest assembly however is. In these models, the affinity is modulating the unbinding rate. Thus, lower affinities yield higher dissociation constants that lead to faster unbinding rates*. If the system with the lowest $K_D$ (slowest unbinding rate) has equilibrated, I am confident

* As discussed in section 7.1.4, a model constructed with the opposite assumption, where the affinity changes modulate the binding rate would observe the opposite behavior.
Figure 7.7: Time traces for the three bond types that structure the β-catenin degradation machinery. APC’s Oligo dimerization (top), APC-Axin recruitment summed for the three SAMP repeats (middle), and Axin’s DIX-head to DIX-tail polymerization (bottom). Systems colored by shared Axin abundance (left) or by shared interactions affinity (right). Time in seconds, abscissa represents molecules, in particle number. Observable patterns defined in appendix C.

the rest of the systems have done so too, hence I present the changes in complex size for the slowest systems in fig. 7.8.

Figure 7.8: Time traces for the size of the largest complex for the systems with \([Axin] = 1000\text{nM} \) and \(K_D = 1000\text{nM}\) (left), \([Axin] = 30\text{nM} \) and \(K_D = 1000\text{nM}\) (middle), and \([Axin] = 1\text{nM} \) and \(K_D = 1000\text{nM}\) (left). Time in seconds.

As for the destruction activity of the system, they also appear at steady state fig. 7.9.
Figure 7.9: Rate of degradation of β-catenin as a function of time. Each degradation event increases a variable, for which the discrete difference in time yields its value change; dividing by the rate of time sampling yields the rate of change in time. Systems colored by shared Axin abundance (left) or by shared interactions affinity (right). Time in seconds, $\Delta x/\Delta t$ in molecules per second.

Based on these observations, I conservatively claim the systems have reached steady state from $T = 40 s$ onward. The results in sections 7.2.3.2 and 7.2.3.3 sample 100 states each, from $T = 40 s$ to $T = 50 s$.

7.2.3.2 What is size of the destruction complex?

In fig. 7.10, the blue plots answer the question “how big are the complexes where Axin is found?”. For all parameter values, the Axin mass in small complexes is negligible. At high abundance (top row), regardless of affinity, the bulk of Axin is present in complexes close to tetramers (4000-mer). These complexes also represent a substantial fraction of the overall mixture. At medium abundances (middle row), regardless of affinity, the bulk of Axin is present in complexes close to dimers (2000-mer). These complexes represent a substantial, albeit reduced, fraction of the overall mixture. At
low abundances (bottom row), regardless of affinity, the bulk of Axin is presented in complexes around the pentacontamers and hectamers (500-mer and 100-mer), however these complexes still represent a small fraction of the overall mixture.

Figure 7.10: Distribution of overall mass and of Axin, for three values of Axin concentration and affinity for the Axin-Axin, APC-APC, and Axin-APC interactions (all three are varied in tandem). Each distribution is averaged over 100 states; systems downsized by 1:100, such that $|\alpha = 10nM \equiv |\alpha = 135$ proteins. Histogram function binned into 100 discrete segments.

Comparing the blue plots to the orange plots in fig. 7.10 answers the question “is the overall complex distribution informative of the Axin distribution?”. Across all parameters, Axin is not in the small-sized fraction of the mixture (e.g. monomers), even if that fraction is dominant in the mixture. Surprisingly, changing the dissociation constant of the three core interactions that underlie the
destruction complex by four orders of magnitude, from \( \mu\text{mol} \) to \( 100\text{pmol} \) has negligible effects on Axin distribution (fig. 7.10). In contrast, changing the concentration of Axin from \( \text{nM} \) to \( \mu\text{M} \) has dramatic effects on Axin distribution. The distribution of Axin in mostly large assemblies, and the sensitivity to protein expression, agree with experimental observations.

Figure 7.11: For the Axin mass distributions (blue) presented in fig. 7.10, these are the cumulative fractions, grouped by similar affinities (top row), or similar abundances (bottom row).

Another plot of the Axin mass distribution is presented in fig. 7.11, this time plotting the cumulative fraction of each parameter pair, and grouping by either similar abundance, or similar affinity. The top row presents the effects of changing the abundance of Axin while keeping the affinities constant. The bottom row presents the effects of changing the affinities while keeping the abundance of Axin constant. Changes in the concentration of Axin has a far more dramatic effect than changing the affinity of the interactions. In other words, the stability of Axin, a phenomenon regulated by Wnt, can have dramatic effects on the size, ergo catalytic potential, of the destruction complex. These changes can easily eclipse the effect of changing the Axin-Axin, Axin-APC, and APC-APC
affinities, another phenomenon regulated by Wnt. As for examples of what such destruction complexes could look like, samples are presented in fig. 7.12.

Figure 7.12 presents contrasting realizations of what can be described as the destruction complex. At low abundances, this is a small complex, easily parsed by eye figs. 7.12g to 7.12i. In contrast, at high abundances this has become a hairball figs. 7.12a to 7.12c. As for the effect of affinity, it has negligible effects over the range explored.

Based on these results, it is not fair to the biology to consider Axin a monomer, or a member of a small complex. Even under limiting concentrations, the presence of other scaffolds and enzymes yields non-trivial complexes. Under higher concentrations, the destruction complex easily becomes a much larger entity, dynamic and pleiomorphic (fig. 7.12).

7.2.3.3 How effective are these complexes?

The capacity to degrade β-catenin tracks well with the size of these complexes, exhibiting greater sensitivity to protein abundance than to affinity values (fig. 7.13).

As previously seen for complex size in fig. 7.12, the system’s capacity to degrade β-catenin is also more sensitive to abundance changes than to affinity changes (fig. 7.13). In fact, regardless of affinity assayed, none of the systems using the low abundance can competently oppose the β-catenin creation rate used throughout the models.

The enzymes responsible for degradation of β-catenin bind reversibly Axin. Each of CK1ε, GSK3β, SCF-βTrCP complex, and the proteasome, all have independent probabilities of being loaded onto any given Axin. Outside of saturating conditions, any specific Axin is unlikely to bind all four en-
tities at the same time. Any specific Axin polymer will therefore have a distribution of recruitment levels. When considering the whole mixture, with Axin forming dynamic polymers, it is clear the destruction of β-catenin could conceivably be carried out on various complexes, with some achieving phosphorylation and others finishing ubiquitination. To explore this, next I quantify the distribution of the four enzymatic activities among the Axin complexes (fig. 7.14).

As one would expect from a regime frequently demixing, fig. 7.14 presents the overwhelming majority of Axins as existing in complexes containing all four enzymatic activities, for all abundances and affinities explored. Let us view the system through the prozone inhibition lens. At low Axin abundance, full recruitment is likely as the bonds are closer to saturation. At high Axin abundance, full recruitment should be unlikely as the system enters the prozone inhibition regime. However, rampant polymerization has abated the inhibitory regime. At low Axin abundances (bottom row), the enzymatic recruitment of any given Axin is highest, promoting full enzymatic load. At high Axin abundances (top row), the system is likely to be so large as to have recruited all activities, even if these seldom land on the same Axin. It is at medium abundances that the greatest diversity of partial assemblies is seen, even if they nonetheless account for a small fraction of the overall Axin mass distribution.
Figure 7.12: These networks represent the largest Axin-containing complex in the final state for key simulations outlined above. Same parameter layout as in fig. 7.10. For the high and medium abundances, the largest Axin-containing complex was the largest complex. For low abundances, the largest complex in the mixture were Dvl fibrils, negative for Axin content.
Figure 7.13: Time traces for evolution of the concentration of β-catenin for the various systems described in fig. 7.12, same layout. The ordinate is Time, in seconds. The coordinate is abundance, in number of proteins.
Figure 7.14: Axin mass classified by the enzymatic types loaded on that complex. Systems as described in figs. 7.10, 7.12 and 7.13, same order. For each plot, along the ordinate axis, tick marks denote in boolean fashion the enzymatic load of complexes. First digit corresponds to the presence of CK1α, second for GSK3β, third for β-TrCP, fourth for the proteasome. Color choice is meant to group by load level, blue for zero activities, orange for one, green for two, red for three, purple for all four. Average over 100 states.
7.2.4 Discussion

In section 7.2.3.1, a careful reader might spot a parameter set that was used in section 7.1.3.1, specifically the model with $[\text{Axin}] = 30\text{nM}$ and $K_D = 10\text{nM}$. These are however not directly comparable. Firstly, their APC abundances are not identical; in section 7.1, the number of SAMP sites is at $30\text{nM}$, which at three per wild-type APC protein, would equate to $10\text{nM}$, vs. the $30\text{nM}$ used in section 7.2. Secondly, the system in section 7.2 is at 1:1 sizing, whereas the system in section 7.1 is at 1:100 downsizing.

In section 7.2.3.2, the high and medium concentration values for Axin track the values reported for Homo epithelial cells, whereas the low value is closer that reported for Xenopus egg. To the extent these number resemble those in Xenopus egg extract vs. Homo cells, the Axin fibrils, and by extension the resulting destruction complex, would be present as smaller entities in Xenopus egg that those found in adult Homo epithelial cells.

In section 7.2.3.3, the systems with low Axin concentrations were unable to competently degrade $\beta$-catenin. That abundance is orders of magnitude higher than that reported in Xenopus egg extract. The realization of such a concentration into a volume on the order of $1\text{pL}$ reaches a floor effect on the number of molecules, thereby limiting the total number of destruction complex species. Using a volume closer to the $1\mu\text{L}$ of the Xenopus egg would contain more Axin agents, by extension allowing a greater diversity of complexes. Even if small, their greater number might be able to competently degrade $\beta$-catenin at the production rate used here. However, at $1.2 \times 10^7$ agents for Axin alone, it would be more costly than any other simulation presented in this work.
7.3 **Take home messages**

- Scaffolds scaffolding scaffolds scaffolded on scaffolds scaffolding other scaffolds can yield phase transitions at a wide range of affinities.
- Scaffold systems can act as programmable locales, where by increasing local concentrations, enzymatic turnovers can be greatly increased.
- Understanding a complex requires understanding its composition, which is comparatively more sensitive to protein abundances than to affinities.
- Mechanisms that change the number of binding opportunities can dramatically alter the size and behavior of complexes.
“As we understand more about the tiniest pieces that we are made of, it becomes increasingly clear that we do not understand how they work together as systems.”

Marc Kirschner, 2003

Biology is complicated. Biological signaling cascades are intricate chains of events. A myriad mechanisms orchestrate a vast array of small molecular changes that together underpin one of the core pillars for multicellular life: cell signaling. Understanding the *whys*, the *hows*, the *whens*, has proven difficult, not least by biology itself resisting formalization.
The Kappa formalism is limited to connections and states; it has no support for geometry, and poor encodings for locality and compartments. And so the model presented here is certainly not the one true model of Wnt signaling. Representing the mechanisms by which β-catenin navigates the nuclear pore complex, by which APC crystalline deposits form, the geometry of polyubiquitin chains and their recruitment by the 26S proteasome, a myriad regulatory steps surrounding Wnt signaling lie well beyond what Kappa can transparently represent. And yet, in spite of these limitations, its level of abstraction is capable of not only recapitulating key aspects of Wnt signaling, but also of explaining them. There are facets of Wnt signaling that can be understood just at the level of complex formation, state modification, and temporal kinetics. Moreover, the platform can serve to identify and refine abstractions, leading to smaller models amenable to analytical approaches (chapter 4).

In this thesis, I have combined quantitative, systematic, and synthetic approaches, through the lens of a formalism, to study Wnt signaling. The model is akin to a synthetic construction of formalized parts, transparent in its quantitative interpretation, integrative of its data. The formality of Kappa allowed tool development, where advances in computer science (chapter 2) and programming (section 3.1) unlocked novel types of analysis.

The advances above allowed me to show that the modeling approach: can be done (chapter 5), facilitates the analysis of specific types of biological complexity (chapter 6), and can serve to explore not only the effect of numeric parameters, but also of architectural ones (chapter 7). When biology deploys three vs. one SAMP repeat per APC, this can regulate the size and turnover of the “destruction complex” (section 7.1). When biology uses affinity changes vs. unfolding of binding sites, this can change how quickly the Dvl fibrills infiltrate the Axin punta (section 5.2.1). When biology uses
flexible vs. rigid components for the assembly of complexes, this can change a trimer into a polymer, which can nucleate further assemblies (section 5.2.2).

Building the model is of value in and of itself, as it forces the user to formalize the mechanism of action. The model is also easily extendable, and can be grown to include other pathways. This offers the possibility of exploring the mechanisms by which cells deal with the rampant cross-talk between pathways.

The factoids have value beyond the model. They can be deployed to other models, or updated as further research clarifies assumptions made. Beyond Kappa however, it is my hope the corpus of factoids that underpin the Wnt model can be formalized themselves, allowing novel schemes for reasoning over our reasoning of biology.

The additive nature of Kappa allows for the instantiating of the model in different contexts. One could develop models that are specific to species, or even to gene expression profiles, conceivably even individual haplotypes. This would facilitate comparative biology at the level of pathway operation, not just gene sequence conservation. Could an individual’s idiosyncratic genotype result in significantly divergent behavior of their canonical Wnt signaling pathway? With the ever cheapening of individual deep sequencing and high throughput proteomics, the framework illustrated in this work could present a mechanistic representation of an individual’s inner workings, critically one that could be “debugged” (e.g. section 6.3). A predictive, mechanistic, interpretable model, that can be inspected, holds great promise for personalized medicine.
Presentation of the Wnt model

INTRODUCTION

This is the presentation of the Wnt model. The goal was a mechanistic model of canonical Wnt signaling, presented in section A.1. Additional interactions that were found but are not part of canonical Wnt signaling are included in section A.2. All results presented above are generated from the interactions in section A.1; interactions in section A.2 are not used, but included merely as a resource.
When displaying Kappa code, a blue background marks operations that change internal state (e.g. phosphorylation), green background marks operations that change bond state (e.g. bind), red background mark operations that change agent abundance (e.g. degradation). Agent names are in bold typeface. Variables are in gray italic typeface. See section 3.2 for details on this rendering.

Unless otherwise stated, these are the default values for variables used in calculating rates:

```plaintext
%var: 'GeneralBinding' 1.0e-3
%var: 'GeneralUnbinding' 1.0e-1
%var: 'GeneralDephosphorylation' 1.0e1
%var: 'GeneralPhosphorylation' 1.0
%var: 'ReceptorDimerization' 1.6e-5
%var: 'ReceptorDissociation' 1.6e-2
%var: 'GeneralUbiquitination' 3.0e-2
%var: 'GeneralDeubiquitination' 1.0
%var: 'GeneralProteolysis' 2.0e1
%var: 'RingClosureRate' 1.0e3
%var: 'Nano_Avo_Vol' 1350
%var: 'RescaleFactor' 0.001
%var: 'EntropyFactor' 0.1
%var: 'Wnt_time' 100
```

To reflect changes in affinity after post-translational modification (e.g. phosphorylation), one can increase the binding rate (vs. unphosphorylated interaction), or decrease the unbinding rate. We chose the latter for efficiency reasons; creating a slower time scale allows us to keep our process of interest as the fast timescale, thereby maintaining its simulatability. The alternate choice can be made where applicable.

The agents we use correspond directly to proteins. E.g. “Fzd” represents a single protein of type Fzd, not a dimer, not a complex with other proteins, but a single protein. The only exceptions are...
for the proteasome and ubiquitination machinery. As the assembly and regulation of those complexes lies outside the scope of this project, each of these entities are encoded as a single agent. I.e. our “proteasome” agent represents a fully assembled proteasome in a configuration suitable for operating on β-catenin and being regulated by the Wnt pathway, idem for the SCFβ-TrCP E3 ligase complex.

These are the initial concentrations for the reference model (in nanomolar):

1. \( \text{ Init}_Dsh3 \) \((100 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
2. \( \text{ Init}_APC \) \((100 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
3. \( \text{ Init}_Axin1 \) \((100 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
4. \( \text{ Init}_KC1A \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
5. \( \text{ Init}_GSK3B \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
6. \( \text{ Init}_BTRCP \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
7. \( \text{ Init}_Proteasome \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
8. \( \text{ Init}_Fzd \) \((50 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
9. \( \text{ Init}_LRP \) \((50 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
10. \( \text{ Init}_PP1A \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
11. \( \text{ Init}_{PP2|B56a} \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
12. \( \text{ Init}_{PP2|B55a} \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
13. \( \text{ Init}_{PP2|Alfa} \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
14. \( \text{ Init}_{PP2|Cata} \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
15. \( \text{ Init}_KC1E \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)
16. \( \text{ Init}_KC1D \) \((10 \times \text{Nano}_Avo\_Vol)\times\text{RescaleFactor}\)

A.1 Canonical Wnt

These are the interactions we found that can suffice to recapitulate “canonical Wnt signaling”. For clarity and narrative ease, the interactions are grouped in three: destruction complex formation and
activities (section A.1.1), signalosome formation and regulation (section A.1.2), and other regulatory events (section A.1.3).

A.1.1 Destruction Complex interactions

A.1.1.1 APC dimerization

APC multimerizes through the Oligo domain\(^48\). Assuming the “coiled-coil” interaction, we interpret this as dimerization rather than higher order polymerization. As this interaction is symmetric, the bi-molecular binding rate is halved.

\[
\begin{align*}
\text{'APC,APC+' } & \text{APC(Oligo[.1]), APC(Oligo[.1]) } @ ((0.5 \times (1 / \text{RescaleFactor})) \\
& \rightarrow \times \text{GeneralBinding}) \{(\text{RingClosureRate} \times \text{GeneralBinding})\} \\
\text{'APC,APC-' } & \text{APC(Oligo[1.]), APC(Oligo[1.]) } @ (0.5 \times \text{GeneralUnbinding})
\end{align*}
\]

A.1.1.2 Axin polymerization

Axin can homopolymerize\(^41\) by concatenating its DIX domain in a head-to-tail manner\(^27\).

\[
\begin{align*}
\text{'Axin,Axin+' } & \text{Axin(DIX-head[.1]), Axin(DIX-tail[.1]) } @ ((1 / \text{RescaleFactor}) \times \text{GeneralBinding}) \{(\text{RingClosureRate} \times \text{GeneralBinding})\} \\
\text{'Axin,Axin-' } & \text{Axin(DIX-head[1.]), Axin(DIX-tail[1.]) } @ \text{GeneralUnbinding}
\end{align*}
\]

A.1.1.3 Recruitment of Axin by APC

APC possesses three binding sites for Axin, the SAMP repeats\(^8\), which bind Axin’s RGS site\(^36\). This interaction is however conditional on the state of co-polymerization of Axin and Dvl, where bind-
ing on Axin of APC is exclusive to binding Dvl, and vice versa. We interpret this as meaning that Axin binding APC can only proceed if that Axin is not bound to a Dvl (i.e. it is either unbound, or bound to an Axin).

1. 'APC(SAMP-1), Axin(DIXh,DIXt)+' APC(SAMP-1[.1]), Axin(RGS[.1] DIX-head[.]) DIX-tail[.] @ ((1 / RescaleFactor) * GeneralBinding)
   → {(RingClosureRate * GeneralBinding)}

2. 'APC(SAMP-2), Axin(DIXh,DIXt)+' APC(SAMP-2[.1]), Axin(RGS[.1] DIX-head[.]) DIX-tail[.] @ ((1 / RescaleFactor) * GeneralBinding)
   → {(RingClosureRate * GeneralBinding)}

3. 'APC(SAMP-3), Axin(DIXh,DIXt)+' APC(SAMP-3[.1]), Axin(RGS[.1] DIX-head[.]) DIX-tail[.] @ ((1 / RescaleFactor) * GeneralBinding)
   → {(RingClosureRate * GeneralBinding)}

4. 'APC(SAMP-1), Axin(DIXh,DIXt!_,DIXt)+' APC(SAMP-1[.1]), Axin(RGS[.1] DIX-head[.] DIX-tail[Axin] DIX-tail[.]) @ ((1 / RescaleFactor) * GeneralBinding)
   → {(RingClosureRate * GeneralBinding)}

5. 'APC(SAMP-2), Axin(DIXh,DIXt!_,DIXt)+' APC(SAMP-2[.1]), Axin(RGS[.1] DIX-head[.] DIX-tail[Axin] DIX-tail[.]) @ ((1 / RescaleFactor) * GeneralBinding)
   → {(RingClosureRate * GeneralBinding)}

6. 'APC(SAMP-3), Axin(DIXh,DIXt!_,DIXt)+' APC(SAMP-3[.1]), Axin(RGS[.1] DIX-head[.] DIX-tail[Axin] DIX-tail[.]) @ ((1 / RescaleFactor) * GeneralBinding)
   → {(RingClosureRate * GeneralBinding)}
By a similar argument, recruitment of Dvl actively ejects APC from Axin.
or if there are additional interactions, if the binding contributes to the Wnt cascade. We take therefore only the 20 amino-acid repeats; extending the model by adding the 15 amino-acid repeats would be straightforward.

The phosphorylation of the repeats increases affinity towards β-catenin, more so if β-catenin is itself phosphorylated, directly tying the phosphostate of the repeat to the capacity of APC to regulate Wnt signaling. From our simulations, the kinetics of phosphorylation on a single repeat are very fast, enough to disregard intermediate phosphostates as extremely short-lived: i.e. a specific repeat will be present in either fully phosphorylated, or fully unphosphorylated forms, with the intermediate phosphostates existing too briefly to be of consequence.

For each repeat, we use a single binding rule, and 4 unbinding rules, iterating over the phosphostate of β-catenin and APC.

1. `'CateninB, APC(rpt20AA1)+' bCat(iARM[./1]), APC(rpt20AA1[./1]) @ ((1 / RescaleFactor) * GeneralBinding)
   2. `'CateninB, APC(rpt20AA2)+' bCat(iARM[./1]), APC(rpt20AA2[./1]) @ ((1 / RescaleFactor) * GeneralBinding)
   3. `'CateninB, APC(rpt20AA3)+' bCat(iARM[./1]), APC(rpt20AA3[./1]) @ ((1 / RescaleFactor) * GeneralBinding)
   4. `'CateninB, APC(rpt20AA4)+' bCat(iARM[./1]), APC(rpt20AA4[./1]) @ ((1 / RescaleFactor) * GeneralBinding)
   5. `'CateninB, APC(rpt20AA5)+' bCat(iARM[./1]), APC(rpt20AA5[./1]) @ ((1 / RescaleFactor) * GeneralBinding)
   6. `'CateninB, APC(rpt20AA6)+' bCat(iARM[./1]), APC(rpt20AA6[./1]) @ ((1 / RescaleFactor) * GeneralBinding)
   7. `'CateninB, APC(rpt20AA7)+' bCat(iARM[./1]), APC(rpt20AA7[./1]) @ ((1 / RescaleFactor) * GeneralBinding)
'CateninB*,APC(rpt20AA1*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA1}[1/.] S1272{ph}[.] S1275{ph}[.] S1276{ph}[.]
  \rightarrow S1278{ph}[.] S1281{ph}[.] ) @ (GeneralUnbinding / 100)

'CateninB*,APC(rpt20AA2*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA2}[1/.] S1386{ph}[.] T1389{ph}[.] S1390{ph}[.]
  \rightarrow S1392{ph}[.] ) @ (GeneralUnbinding / 100)

'CateninB*,APC(rpt20AA3*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA3}[1/.] S1501{ph}[.] S1504{ph}[.] S1505{ph}[.]
  \rightarrow S1507{ph}[.] S1510{ph}[.] ) @ (GeneralUnbinding / 100)

'CateninB*,APC(rpt20AA4*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA4}[1/.] S1652{ph}[.] T1655{ph}[.] S1656{ph}[.]
  \rightarrow S1658{ph}[.] T1661{ph}[.] S1664{ph}[.] ) @ (GeneralUnbinding / 100)

'CateninB*,APC(rpt20AA5*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA5}[1/.] S1857{ph}[.] S1861{ph}[.] S1863{ph}[.]
  \rightarrow S1857{ph}[.] ) @ (GeneralUnbinding / 100)

'CateninB*,APC(rpt20AA6*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA6}[1/.] S1964{ph}[.] S1967{ph}[.] S1968{ph}[.]
  \rightarrow S1964{ph}[.] S1967{ph}[.] S1968{ph}[.] ) @ (GeneralUnbinding / 100)

'CateninB*,APC(rpt20AA7*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA7}[1/.] S2022{ph}[.] S2025{ph}[.] S2026{ph}[.]
  \rightarrow S2028{ph}[.] S2031{ph}[.] S2034{ph}[.] ) @ (GeneralUnbinding / 100)

'CateninB*,APC(rpt20AA8*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA8}[1/.] S2172{un}[.] S2175{un}[.] S2176{un}[.]
  \rightarrow S2172{un}[.] S2175{un}[.] S2176{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA9*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA9}[1/.] S2272{un}[.] S2275{un}[.] S2276{un}[.]
  \rightarrow S2272{un}[.] S2275{un}[.] S2276{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA10*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA10}[1/.] S2372{un}[.] S2375{un}[.] S2376{un}[.]
  \rightarrow S2372{un}[.] S2375{un}[.] S2376{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA11*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA11}[1/.] S2472{un}[.] S2475{un}[.] S2476{un}[.]
  \rightarrow S2472{un}[.] S2475{un}[.] S2476{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA12*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA12}[1/.] S2572{un}[.] S2575{un}[.] S2576{un}[.]
  \rightarrow S2572{un}[.] S2575{un}[.] S2576{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA13*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA13}[1/.] S2672{un}[.] S2675{un}[.] S2676{un}[.]
  \rightarrow S2672{un}[.] S2675{un}[.] S2676{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA14*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA14}[1/.] S2772{un}[.] S2775{un}[.] S2776{un}[.]
  \rightarrow S2772{un}[.] S2775{un}[.] S2776{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA15*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA15}[1/.] S2872{un}[.] S2875{un}[.] S2876{un}[.]
  \rightarrow S2872{un}[.] S2875{un}[.] S2876{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA16*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA16}[1/.] S2972{un}[.] S2975{un}[.] S2976{un}[.]
  \rightarrow S2972{un}[.] S2975{un}[.] S2976{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA17*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA17}[1/.] S3072{un}[.] S3075{un}[.] S3076{un}[.]
  \rightarrow S3072{un}[.] S3075{un}[.] S3076{un}[.] ) @ (GeneralUnbinding / 10)

'CateninB*,APC(rpt20AA18*)- bCat(S33{ph}[.] S37{ph}[.] T41{ph}[.]
  \text{iARM}[1/.], \text{APC(rpt20AA18}[1/.] S3172{un}[.] S3175{un}[.] S3176{un}[.]
  \rightarrow S3172{un}[.] S3175{un}[.] S3176{un}[.] ) @ (GeneralUnbinding / 10)
'CateninB^,APC(rpt20AA5^)-' bCat(S33[ph].. S37[ph].. T41[ph]..)
  iARM[1/..], APC(rpt20AA5[1/..] S1857{un}[..] S1861{un}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA6^)-' bCat(S33[ph].. S37[ph].. T41[ph]..)
  iARM[1/..], APC(rpt20AA6[1/..] S1964{un}[..] S1967{un}[..] S1968{un}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA7^)-' bCat(S33[ph].. S37[ph].. T41[ph]..)
  iARM[1/..], APC(rpt20AA7[1/..] S2022{un}[..] S2025{un}[..] S2026{un}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA1^)-' bCat(S33[un].. S37[un].. T41[un]..)
  iARM[1/..], APC(rpt20AA1[1/..] S1272{ph}[..] S1275{ph}[..] S1276{ph}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA2^)-' bCat(S33[un].. S37[un].. T41[un]..)
  iARM[1/..], APC(rpt20AA2[1/..] S1386{ph}[..] T1389{ph}[..] S1390{ph}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA3^)-' bCat(S33[un].. S37[un].. T41[un]..)
  iARM[1/..], APC(rpt20AA3[1/..] S1501{ph}[..] S1504{ph}[..] S1505{ph}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA4^)-' bCat(S33[un].. S37[un].. T41[un]..)
  iARM[1/..], APC(rpt20AA4[1/..] S1652{ph}[..] T1655{ph}[..] S1656{ph}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA5^)-' bCat(S33[un].. S37[un].. T41[un]..)
  iARM[1/..], APC(rpt20AA5[1/..] S1857{ph}[..] S1861{ph}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA6^)-' bCat(S33[un].. S37[un].. T41[un]..)
  iARM[1/..], APC(rpt20AA6[1/..] S1964{ph}[..] S1967{ph}[..] S1968{ph}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA7^)-' bCat(S33[un].. S37[un].. T41[un]..)
  iARM[1/..], APC(rpt20AA7[1/..] S2022{ph}[..] S2025{ph}[..] S2026{ph}[..] )@ (GeneralUnbinding / 10)

'CateninB^,APC(rpt20AA1^)-' bCat(S33[un].. S37[un].. T41[un]..)
  iARM[1/..], APC(rpt20AA1[1/..] S1272{un}[..] S1275{un}[..] S1276{un}[..] )@ GeneralUnbinding
A.1.1.5 Channeling of β-catenin to APC from Axin

Axin binds β-catenin’s ARM repeats 1 through 5. An APC’s 20 amino-acid repeat can bind repeats 5 through 9 if it is unphosphorylated, or repeats 1 through 9 if it is phosphorylated. This presents a mechanism where, assuming everything is phosphorylated, APC could bind a β-catenin at repeats 5-9 simultaneously with Axin being bound at repeats 1-9, and displace Axin, using the increased affinity as a ratchet, suggested by the model in. We reflect this phenomenon as a bond change for the phosphorylated versions of these proteins.
\[\text{CateninB, Axin(), APC(rpt20AA1*)} \rightarrow \text{bCat}(\text{S33[.]} \text{ S37[.]} \text{ T41[.]} \text{iARM[.]} \text{CBD.}} \text{Axin/1]}], \text{APC(rpt20AA1[]./1]} \text{S1272[.]} \text{S1275[.]} \text{)} @ 0.\]

\{(\text{RingClosureRate} \ast \text{GeneralBinding})\}\]

\[\text{CateninB, Axin(), APC(rpt20AA2*)} \rightarrow \text{bCat}(\text{S33[.]} \text{ S37[.]} \text{ T41[.]} \text{iARM[.]} \text{CBD.}} \text{Axin/1]}], \text{APC(rpt20AA2[]./1]} \text{S1386[.]} \text{T1389[.]} \text{)} @ 0.\]

\{(\text{RingClosureRate} \ast \text{GeneralBinding})\}\]

\[\text{CateninB, Axin(), APC(rpt20AA3*)} \rightarrow \text{bCat}(\text{S33[.]} \text{ S37[.]} \text{ T41[.]} \text{iARM[.]} \text{CBD.}} \text{Axin/1]}], \text{APC(rpt20AA3[]./1]} \text{S1504[.]} \text{)} @ 0.\]

\{(\text{RingClosureRate} \ast \text{GeneralBinding})\}\]

\[\text{CateninB, Axin(), APC(rpt20AA4*)} \rightarrow \text{bCat}(\text{S33[.]} \text{ S37[.]} \text{ T41[.]} \text{iARM[.]} \text{CBD.}} \text{Axin/1]}], \text{APC(rpt20AA4[]./1]} \text{S1664[.]} \text{)} @ 0.\]

\{(\text{RingClosureRate} \ast \text{GeneralBinding})\}\]

\[\text{CateninB, Axin(), APC(rpt20AA5*)} \rightarrow \text{bCat}(\text{S33[.]} \text{ S37[.]} \text{ T41[.]} \text{iARM[.]} \text{CBD.}} \text{Axin/1]}], \text{APC(rpt20AA5[]./1]} \text{S1863[.]} \text{)} @ 0.\]

\{(\text{RingClosureRate} \ast \text{GeneralBinding})\}\]

\[\text{CateninB, Axin(), APC(rpt20AA6*)} \rightarrow \text{bCat}(\text{S33[.]} \text{ S37[.]} \text{ T41[.]} \text{iARM[.]} \text{CBD.}} \text{Axin/1]}], \text{APC(rpt20AA6[]./1]} \text{S1973[.]} \text{)} @ 0.\]

\{(\text{RingClosureRate} \ast \text{GeneralBinding})\}\]

\[\text{CateninB, Axin(), APC(rpt20AA7*)} \rightarrow \text{bCat}(\text{S33[.]} \text{ S37[.]} \text{ T41[.]} \text{iARM[.]} \text{CBD.}} \text{Axin/1]}], \text{APC(rpt20AA7[]./1]} \text{S2034[.]} \text{)} @ 0.\]

\{(\text{RingClosureRate} \ast \text{GeneralBinding})\}\]

\[\text{CateninB, Axin(), APC(rpt20AA1*)} \rightarrow \text{bCat}(\text{S33[.]} \text{ S37[.]} \text{ T41[.]} \text{iARM[.]} \text{CBD.}} \text{Axin/1]}], \text{APC(rpt20AA1[]./1]} \text{S1272[.]} \text{S1275[.]} \text{)} @ 0.\]

\{(\text{RingClosureRate} \ast \text{GeneralBinding}) / 10)\}\]
A.1.6 Recruitment of β-TrCP complex by Axin

The β-TrCP complex has some association with the destruction complex\(^68\), though this is not done through APC and may be through Axin\(^51\). Through a complex involving YAP and TAZ, it is brought in contact with Axin. LRP appears to compete against the YAP- TAZ entities, possibly displacing β-
TrCP from the Axin.

As the point of regulation remains the binding site on Axin, and YAP and TAZ are not canonical members of the canonical pathway, we are not representing them directly. Rather, our notion of “effective SCF β-TrCP” now includes the appropriate YAP-TAZ forms, capable of delivering E3 ligase activity to the destruction complex. The possible competition by LRP is also not represented, as that mechanism is unclear.

A.1.1.7 Recruitment of 26S proteasome by Axin

Axin binds two subunits of the proteasome, PSMD2 and PSMD6. These are subunits of the 19S regulatory particle, therefore we refer to the 26S proteasome.

A.1.1.8 Binding of Axin and β-catenin

Phosphorylation of Axin increases its affinity for β-catenin. The sites in question are the ones that lie within the β-catenin binding site.
A.1.1.9 Interaction between β-catenin and the β-TrCP complex

β-catenin, phosphorylated at Serine 33 and Serine 37, is recognized by the β-TrCP complex. As the complex is physically brought into proximity through other means, this interaction concerns the location of ubiquitination on β-catenin as it falls into the catalytic site of the β-TrCP complex. The release of the segment containing the appropriate Lysine will happen preferentially if the Lysine is ubiquitinated.
A.1.1.10 Recruitment of \( \beta \)-catenin by the 26S proteasome

\( \beta \)-catenin that is ubiquitinated will be recognized and engaged by the 26S proteasome\(^1\). We assume this binding is not an irreversible reaction, even if the kinetics of degradation make it fast enough to rarely observe unbinding. As for the binding site, although \( \beta \)-catenin’s ARM repeats offer a prime dock, we consider it more conservative to assume the binding occurs through the ubiquitin chain attached to \( \beta \)-catenin, as that is the molecular entity recognized by the proteasome.

\[
\text{'CateninB#,Proteasome+':} \quad \text{bCat}(\text{LysX\{ub\}[.1]), Prot(CatDom[.1])} @ ((1 / \rightarrow \text{RescaleFactor}) \ast \text{GeneralBinding}) \{(\text{RingClosureRate} \ast \text{GeneralBinding})\}
\]

\[
\text{'CateninB#,Proteasome-':} \quad \text{bCat}(\text{LysX\{ub\}[1./1]), Prot(CatDom[1./1])} @ \rightarrow \text{GeneralUnbinding}
\]

A.1.1.11 Phosphorylation, ubiquitination, and degradation of \( \beta \)-catenin

\( \beta \)-catenin is phosphoprimed at Serine 45 by CK1\( \alpha \). Once S45 primed, it is phosphorylated at Threonine 41 by GSK3\( \beta \). Once T41 phosphorylated, it is hyperphosphorylated by GSK3\( \beta \) at Serines 37 and 33. Once hyperphosphorylated at S37 and S33, it is ubiquitinated at an unknown Lysine by an SCF \( \beta \)-TrCP complex\(^5\). Once ubiquitinated\(^1\), it is directly degraded by the 26S proteasome\(^1\), after binding it directly\(^7\).

The phosphorylation and ubiquitination reactions are written using the unary rate to fire events, rather than the binary one which would translate to action at a distance. This is due to the rampant disorder and polymerization found in the system. To us, it is sensible to consider the case where a \( \beta \)-catenin bound on one Axin can be phosphorylated by the CK1\( \alpha \) bound on the neighboring Axin,
idem for ubiquitination. However, for proteosomal degradation, we do not make this assumption
about mechanism, as β-catenin’s ubiquitin is physically interacting with the regulatory particle of the
proteasome.

'A.1.1.12  Phosphorylation of Axin by CK1α and GSK3β

CK1α phosphorylates Axin on 4 residues, the first 3 associated with Axin stability,\(^{62}\), the fourth ly-
ing withing the β-catenin binding site. GSK3β phosphorylates Axin\(^{93}\) on 2 residues that affect the
affinity to β-catenin, both lying within the β-catenin binding site\(^{90}\).

There is a discrepancy between the residue positions identified above and those presented here;
we are using the indexing as recorded at UniProt entry AXIN1_HUMAN, which borrows “by similarity” the entries from AXIN1_MOUSE, which cites the above papers.

A.1.1.13 Recruitment of CK1[α, δ, ε] by Axin

Axin directly binds several members of the CK1 family, specifically CK1α, CK1δ, and CK1ɛ.\(^3, 59, 98\).

We are aware CK1δ is a membrane protein, so the binding to cytosolic Axin would be rare; that said, can CK1δ act as an anchor, stabilizing Axin at the membrane? Even if the interaction is not strong enough by itself, once Axin is membrane-associated via other proteins, it is conceivable CK1δ strengthen this localization. As there is little data on protein abundances at the membrane, or kinetics of transport to the membrane, we are not modeling the compartment aspect.

1  'KC1α().Axin(S75)*' Axin(S75[un/ph][.]), CK1α()  
   ↦ 1 Axn−75-p @ 0. {GeneralPhosphorylation}

2  'KC1α().Axin(S77)*' Axin(S77[un/ph][.]), CK1α()  
   ↦ 1 Axn−77-p @ 0. {GeneralPhosphorylation}

3  'KC1α().Axin(S217)*' Axin(S217[un/ph][.]),  
   ↞ CK1α()  
   ↦ {GeneralPhosphorylation}  
   ↦ 1 Axn−217-p @ 0.

4  'KC1α().Axin(S469)*' Axin(CBD[.] S469[un/ph][.]), CK1α()  
   ↞ Axn−469-p @ 0. {GeneralPhosphorylation}

5  'GSK3().Axin(T481)*' Axin(CBD[.] T481[un/ph][.]), GSK()  
   ↞ Axn−481-p @ 0. {GeneralPhosphorylation}

6  'GSK3().Axin(S493)*' Axin(CBD[.] S493[un/ph][.]), GSK()  
   ↞ Axn−493-p @ 0. {GeneralPhosphorylation}
A.1.1.14  GSK₃β recruitment by Axin

GSK₃β binds Axin, with increasing affinity as Axin is phosphorylated on certain residues.

'GSK₃β, Axin+': GSK(Axin[1/1]), Axin(GSK[1/1]) @ ((1 / RescaleFactor) * GeneralBinding)

'GSK₃β, Axin(****)-': GSK(Axin[1/1]), Axin(GSK[1/1]) S75{ph}[] S77{ph}[] S217{ph}[] S469{ph}[] @ (GeneralUnbinding / 10)

'GSK₃β, Axin(***^)-': GSK(Axin[1/1]), Axin(GSK[1/1]) S75{ph}[] S77{ph}[] S217{ph}[] S469{un}[] @ (GeneralUnbinding * 8)

'GSK₃β, Axin(^***)-': GSK(Axin[1/1]), Axin(GSK[1/1]) S75{ph}[] S77{un}[] S217{ph}[] S469{ph}[] @ (GeneralUnbinding * 8)

'GSK₃β, Axin(**^)-': GSK(Axin[1/1]), Axin(GSK[1/1]) S75{un}[] S77{ph}[] S217{ph}[] S469{ph}[] @ (GeneralUnbinding * 8)

'GSK₃β, Axin(*^^)-': GSK(Axin[1/1]), Axin(GSK[1/1]) S75{un}[] S77{un}[] S217{ph}[] S469{un}[] @ (GeneralUnbinding * 4)

'GSK₃β, Axin(^**^)-': GSK(Axin[1/1]), Axin(GSK[1/1]) S75{un}[] S77{un}[] S217{un}[] S469{ph}[] @ (GeneralUnbinding * 4)
A.1.1.15 Phosphorylation of APC by CK\(\varepsilon\) and GSK\(3\beta\)

Each of the 20 amino-acid repeats has residues capable of being phosphorylated, or that are outright phosphomimetic, in a recurring pattern\(^{71,34}\). In this section, we refer to these as sites \(\alpha, \beta, \gamma, \delta, \varepsilon, \) and \(\zeta\), in alphabetical order following their precedence; in the model they are named after their amino-acid and sequence position. Sites \(\alpha\) match the phosphorylation pattern of CK\(\varepsilon\) for all repeats. With sites \(\alpha\) primed, sites \(\beta\) match the phosphorylation pattern by GSK\(3\beta\), for all repeats. With sites \(\beta\) primed, sites \(\gamma\) match the phosphorylation pattern by CK\(\varepsilon\) for all repeats sauf 5, which carries a phosphomimetic aspartic acid. With sites \(\gamma\) primed (or phosphomimicking), sites \(\delta\) match the phosphorylation pattern by CK\(\varepsilon\) for all repeats. With sites \(\delta\) primed, sites \(\varepsilon\) match the phosphorylation pattern by CK\(\varepsilon\) for all repeats, sauf repeats 2 and 5, which carry phosphomimicking aspartic
acids. With sites $\varepsilon$ primed (or phosphomimicking), sites $\zeta$ match the phosphorylation pattern by CK$\varepsilon$ for repeat 7, as all other repeats carry a phosphomimicking aspartic or glutamic acid.

<table>
<thead>
<tr>
<th>Order</th>
<th>$\text{CK}_\varepsilon \rightarrow \alpha$</th>
<th>$\text{GSK}_\beta \rightarrow \beta$</th>
<th>$\text{CK}_\gamma \rightarrow \gamma$</th>
<th>$\text{CK}_\delta \rightarrow \delta$</th>
<th>$\text{CK}_\varepsilon \rightarrow \varepsilon$</th>
<th>$\text{CK}_\zeta \rightarrow \zeta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site</td>
<td>$\beta$</td>
<td>$\gamma$</td>
<td>$\alpha$</td>
<td>$\delta$</td>
<td>$\varepsilon$</td>
<td>$\zeta$</td>
</tr>
<tr>
<td>rpt1</td>
<td>S1272</td>
<td>S1275</td>
<td>S1276</td>
<td>S1278</td>
<td>S1281</td>
<td>E1284</td>
</tr>
<tr>
<td>rpt2</td>
<td>S1386</td>
<td>T1389</td>
<td>S1390</td>
<td>S1392</td>
<td>D1395</td>
<td>E1398</td>
</tr>
<tr>
<td>rpt3</td>
<td>S1501</td>
<td>S1504</td>
<td>S1505</td>
<td>S1507</td>
<td>S1510</td>
<td>E1513</td>
</tr>
<tr>
<td>rpt4</td>
<td>S1652</td>
<td>T1655</td>
<td>S1656</td>
<td>S1658</td>
<td>T1661</td>
<td>S1664</td>
</tr>
<tr>
<td>rpt5</td>
<td>S1857</td>
<td>D1860</td>
<td>S1861</td>
<td>S1863</td>
<td>D1866</td>
<td>D1969</td>
</tr>
<tr>
<td>rpt7</td>
<td>S2022</td>
<td>S2025</td>
<td>S2026</td>
<td>S2028</td>
<td>S2031</td>
<td>S2034</td>
</tr>
</tbody>
</table>

Table A.1: Composition of each repeat of APC in terms of phosphorylated sites, or negatively charged residues. Residues sorted by amino-acid position, grouped by repeat into rows. The phosphorylation order is presented at the top row of the table for reference purposes, along with the enzyme responsible for that reaction: $A \xrightarrow{B} C$ means B phosphorylates C once A has been phosphorylated.
'KC1e^().APC(S1281)*' APC(S1278[ph][.] S1281[un/ph][.]), CK1e(S323[ph][.]
⇒ T325[ph][.] T334[ph][.] T337[ph][.] S368[ph][.] S405[ph][.]
⇒ T407[ph][.] S408[ph][.]) | 1 APC1281-p @ 0.
⇒ {(GeneralPhosphorylation / 10)}

// APC 20aaR2: S1386, T1389, S1390, S1392, D1395, E1398
'KC1e^().APC(S1390)*' APC(S1390[un/ph][.]), CK1e(S323[ph][.] T325[ph][.]
⇒ T334[ph][.] T337[ph][.] S368[ph][.] S405[ph][.] T407[ph][.]
⇒ S408[ph][.]) | 1 APC1390-p @ 0.
⇒ {(GeneralPhosphorylation / 10)}

'GSK3().APC(S1386)*' APC(S1386[un/ph][.] S1390[ph][.]),
⇒ GSK()
⇒ 1 APC1386-p @ 0. {GeneralPhosphorylation}

'KC1e^().APC(T1389)*' APC(S1386[ph][.] T1389[un/ph][.]), CK1e(S323[ph][.]
⇒ T325[ph][.] T334[ph][.] T337[ph][.] S368[ph][.] S405[ph][.]
⇒ T407[ph][.] S408[ph][.]) | 1 APC1389-p @ 0.
⇒ {(GeneralPhosphorylation / 10)}

'KC1e^().APC(S1392)*' APC(T1389[ph][.] S1392[un/ph][.]), CK1e(S323[ph][.]
⇒ T325[ph][.] T334[ph][.] T337[ph][.] S368[ph][.] S405[ph][.]
⇒ T407[ph][.] S408[ph][.]) | 1 APC1392-p @ 0.
⇒ {(GeneralPhosphorylation / 10)}

// APC 20aaR3: S1501, S1504, S1505, S1507, S1510, E1513
'KC1e^().APC(S1505)*' APC(S1505[un/ph][.]), CK1e(S323[ph][.] T325[ph][.]
⇒ T334[ph][.] T337[ph][.] S368[ph][.] S405[ph][.] T407[ph][.]
⇒ S408[ph][.]) | 1 APC1505-p @ 0.
⇒ {(GeneralPhosphorylation / 10)}

'GSK3().APC(S1501)*' APC(S1501[un/ph][.] S1505[ph][.]),
⇒ GSK()
⇒ 1 APC1501-p @ 0. {GeneralPhosphorylation}

'KC1e^().APC(S1504)*' APC(S1501[ph][.] S1504[un/ph][.]), CK1e(S323[ph][.]
⇒ T325[ph][.] T334[ph][.] T337[ph][.] S368[ph][.] S405[ph][.]
⇒ T407[ph][.] S408[ph][.]) | 1 APC1504-p @ 0.
⇒ {(GeneralPhosphorylation / 10)}
'KC1e^().APC(S1507)*'  APC(S1504{ph}.. S1507{un/ph}..), CK1e(S323{ph}..)
   T325{ph}.. T334{ph}.. T337{ph}.. S368{ph}.. S405{ph}..
   T407{ph}.. S408{ph}..)  | 1 APC1507-p @ 0.
   {(GeneralPhosphorylation / 10)}

'KC1e^().APC(S1510)*'  APC(S1507{ph}.. S1510{un/ph}..), CK1e(S323{ph}..)
   T325{ph}.. T334{ph}.. T337{ph}.. S368{ph}.. S405{ph}..
   T407{ph}.. S408{ph}..)  | 1 APC1510-p @ 0.
   {(GeneralPhosphorylation / 10)}

// APC 20aaR4: S1652, T1655, S1656, S1658, T1661, S1664

'KC1e^().APC(S1652)*'  APC(S1656{un/ph}..), CK1e(S323{ph}..) T325{ph}..
   T334{ph}.. T337{ph}.. S368{ph}.. S405{ph}.. T407{ph}..
   S408{ph}..)  | 1 APC1652-p @ 0.
   {(GeneralPhosphorylation / 10)}

'KC1e^().APC(S1656)*'  APC(S1656{un/ph}..), CK1e(S323{ph}..) T325{ph}..
   T334{ph}.. T337{ph}.. S368{ph}.. S405{ph}.. T407{ph}..
   S408{ph}..)  | 1 APC1656-p @ 0.
   {(GeneralPhosphorylation / 10)}

'GSK3().APC(S1652)*'  APC(S1652{un/ph}.. S1656{ph}..),
   GSK()
   1 APC1652-p @ 0. {GeneralPhosphorylation}

'KC1e^().APC(T1655)*'  APC(S1652{ph}.. T1655{un/ph}..), CK1e(S323{ph}..)
   T325{ph}.. T334{ph}.. T337{ph}.. S368{ph}.. S405{ph}..
   T407{ph}.. S408{ph}..)  | 1 APC1655-p @ 0.
   {(GeneralPhosphorylation / 10)}

'KC1e^().APC(S1658)*'  APC(T1655{ph}.. S1658{un/ph}..), CK1e(S323{ph}..)
   T325{ph}.. T334{ph}.. T337{ph}.. S368{ph}.. S405{ph}..
   T407{ph}.. S408{ph}..)  | 1 APC1658-p @ 0.
   {(GeneralPhosphorylation / 10)}

'KC1e^().APC(T1661)*'  APC(S1658{ph}.. T1661{un/ph}..), CK1e(S323{ph}..)
   T325{ph}.. T334{ph}.. T337{ph}.. S368{ph}.. S405{ph}..
   T407{ph}.. S408{ph}..)  | 1 APC1661-p @ 0.
   {(GeneralPhosphorylation / 10)}

'KC1e^().APC(S1664)*'  APC(T1661{ph}.. S1664{un/ph}..), CK1e(S323{ph}..)
   T325{ph}.. T334{ph}.. T337{ph}.. S368{ph}.. S405{ph}..
   T407{ph}.. S408{ph}..)  | 1 APC1664-p @ 0.
   {(GeneralPhosphorylation / 10)}

// APC 20aaR5: S1857, D1860, S1861, S1863, D1866, D1969

221
'KC1e^.().APC(S1861)*'  APC(S1861[un/ph][.]), CK1e(S323{ph}[.]) T325{ph}[.]
→ T334{ph}[.], T337{ph}[.], S368{ph}[.], S405{ph}[.], T407{ph}[.]
→ S408{ph}[.])  | 1 APC1861-p @ 0.
→ {GeneralPhosphorylation / 10}

'GSK3.().APC(S1857)*'  APC(S1857[un/ph][.], S1861{ph}[.]), GSK()
→ 1 APC1857-p @ 0. {GeneralPhosphorylation}

'KC1e^.().APC(S1863)*'  APC(S1863[un/ph][.]), CK1e(S323{ph}[.]) T325{ph}[.]
→ T334{ph}[.], T337{ph}[.], S368{ph}[.], S405{ph}[.], T407{ph}[.]
→ S408{ph}[.])  | 1 APC1863-p @ 0.
→ {GeneralPhosphorylation / 10}


'KC1e^.().APC(S1968)*'  APC(S1968[un/ph][.]), CK1e(S323{ph}[.]) T325{ph}[.]
→ T334{ph}[.], T337{ph}[.], S368{ph}[.], S405{ph}[.], T407{ph}[.]
→ S408{ph}[.])  | 1 APC1968-p @ 0.
→ {GeneralPhosphorylation / 10}

'GSK3.().APC(S1964)*'  APC(S1964[un/ph][.], S1968{ph}[.]), GSK()
→ 1 APC1964-p @ 0. {GeneralPhosphorylation}

'KC1e^.().APC(S1967)*'  APC(S1964{ph}[.], S1967[un/ph][.]), CK1e(S323{ph}[.])
→ T325{ph}[.], T334{ph}[.], T337{ph}[.], S368{ph}[.], S405{ph}[.]
→ T407{ph}[.], S408{ph}[.])  | 1 APC1967-p @ 0.
→ {GeneralPhosphorylation / 10}

'KC1e^.().APC(S1970)*'  APC(S1967{ph}[.], S1970[un/ph][.]), CK1e(S323{ph}[.])
→ T325{ph}[.], T334{ph}[.], T337{ph}[.], S368{ph}[.], S405{ph}[.]
→ T407{ph}[.], S408{ph}[.])  | 1 APC1970-p @ 0.
→ {GeneralPhosphorylation / 10}

'KC1e^.().APC(S1973)*'  APC(S1970{ph}[.], S1973[un/ph][.]), CK1e(S323{ph}[.])
→ T325{ph}[.], T334{ph}[.], T337{ph}[.], S368{ph}[.], S405{ph}[.]
→ T407{ph}[.], S408{ph}[.])  | 1 APC1973-p @ 0.
→ {GeneralPhosphorylation / 10}

// APC 20aaR7: S2022, S2025, S2026, S2028, S2031, S2034
'KC1e^.().APC(S2026)*' APC(S2026{un/ph}[]], CK1e(S323{ph}[]] T325{ph}[]]
   \[ T334{ph}[]] T337{ph}[]] S368{ph}[]] S405{ph}[]] T407{ph}[]]
   \[ S408{ph}[]]\] | 1 APC2026-p @ 0.
   \{(GeneralPhosphorylation / 10)\}

'GSK3.().APC(S2022)*' APC(S2022{un/ph}[]] S2026{ph}[]], GSK()
   \[ 1 APC2022-p @ 0. \{GeneralPhosphorylation\}

'KC1e^.().APC(S2025)*' APC(S2025{ph}[]] S2025{un/ph}[]], CK1e(S323{ph}[]]
   \[ T325{ph}[]] T334{ph}[]] T337{ph}[]] S368{ph}[]] S405{ph}[]]
   \[ T407{ph}[]] S408{ph}[]] | 1 APC2025-p @ 0.
   \{(GeneralPhosphorylation / 10)\}

'KC1e^.().APC(S2028)*' APC(S2028{ph}[]] S2028{un/ph}[]], CK1e(S323{ph}[]]
   \[ T325{ph}[]] T334{ph}[]] T337{ph}[]] S368{ph}[]] S405{ph}[]]
   \[ T407{ph}[]] S408{ph}[]] | 1 APC2028-p @ 0.
   \{(GeneralPhosphorylation / 10)\}

'KC1e^.().APC(S2031)*' APC(S2031{ph}[]] S2031{un/ph}[]], CK1e(S323{ph}[]]
   \[ T325{ph}[]] T334{ph}[]] T337{ph}[]] S368{ph}[]] S405{ph}[]]
   \[ T407{ph}[]] S408{ph}[]] | 1 APC2031-p @ 0.
   \{(GeneralPhosphorylation / 10)\}

'KC1e^.().APC(S2034)*' APC(S2034{ph}[]] S2034{un/ph}[]], CK1e(S323{ph}[]]
   \[ T325{ph}[]] T334{ph}[]] T337{ph}[]] S368{ph}[]] S405{ph}[]]
   \[ T407{ph}[]] S408{ph}[]] | 1 APC2034-p @ 0.
   \{(GeneralPhosphorylation / 10)\}

// For CK1e phosphorylated
// APC 20aaaR1: S1272, S1275, S1276, S1278, S1281, E1284

'KC1e*.().APC(S1276)*' APC(S1276{un/ph}[]], CK1e(S323{un}[]] T325{un}[]]
   \[ T334{un}[]] T337{un}[]] S368{un}[]] S405{un}[]] T407{un}[]]
   \[ S408{un}[]]\] | 1 APC1276-p @ 0.
   \{GeneralPhosphorylation\}

'KC1e*.().APC(S1275)*' APC(S1275{un/ph}[]], CK1e(S323{un}[]]
   \[ T325{un}[]] T334{un}[]] T337{un}[]] S368{un}[]] S405{un}[]]
   \[ T407{un}[]] S408{un}[]] | 1 APC1275-p @ 0.
   \{GeneralPhosphorylation\}
KC1e*.().APC(S1278)* APC(S1275{ph}.. S1278{un/ph}..), CK1ε(S323{un}..)
 T325{un}.. T334{un}.. T337{un}.. S368{un}.. S405{un}..
 T407{un}.. S408{un}..) | 1 APC1278-p @ 0.
 {GeneralPhosphorylation}

KC1e*.().APC(S1281)* APC(S1278{ph}.. S1281{un/ph}..), CK1ε(S323{un}..)
 T325{un}.. T334{un}.. T337{un}.. S368{un}.. S405{un}..
 T407{un}.. S408{un}..) | 1 APC1281-p @ 0.
 {GeneralPhosphorylation}

APC 20aaR2: S1386, T1389, S1390, S1392, D1395, E1398

KC1e*.().APC(S1390)* APC(S1390{un/ph}..), CK1ε(S323{un}.. T325{un}..)
 T334{un}.. T337{un}.. S368{un}.. S405{un}.. T407{un}..
 S408{un}..) | 1 APC1390-p @ 0.
 {GeneralPhosphorylation}

KC1e*.().APC(T1389)* APC(S1386{ph}.. T1389{un/ph}..), CK1ε(S323{un}..)
 T325{un}.. T334{un}.. T337{un}.. S368{un}.. S405{un}..
 T407{un}.. S408{un}..) | 1 APC1389-p @ 0.
 {GeneralPhosphorylation}

APC 20aaR3: S1501, S1504, S1505, S1507, S1510, E1513

KC1e*.().APC(S1504)* APC(S1505{un/ph}..), CK1ε(S323{un}.. T325{un}..
 T334{un}.. T337{un}.. S368{un}.. S405{un}.. T407{un}..
 S408{un}..) | 1 APC1504-p @ 0.
 {GeneralPhosphorylation}

KC1e*.().APC(S1504)* APC(S1501{ph}.. S1504{un/ph}..), CK1ε(S323{un}..)
 T325{un}.. T334{un}.. T337{un}.. S368{un}.. S405{un}..
 T407{un}.. S408{un}..) | 1 APC1501-p @ 0.
 {GeneralPhosphorylation}

KC1e*.().APC(S1507)* APC(S1504{ph}.. S1507{un/ph}..), CK1ε(S323{un}..
 T325{un}.. T334{un}.. T337{un}.. S368{un}.. S405{un}..
 T407{un}.. S408{un}..) | 1 APC1507-p @ 0.
 {GeneralPhosphorylation}
KC1e*.().APC(S1510)* APC(S1507[un]/ph[.]) S1510[un/ph[.]], CK1e(S323[un][.])
→ T325[un][.] T334[un][.] T337[un][.] S368[un][.] S405[un][.]
→ T407[un][.] S408[un][.]) | 1 APC1510-p @ 0.
→ {GeneralPhosphorylation}

// APC 20aaR4: S1652, T1655, S1656, S1658, T1661, S1664
KC1e*.().APC(S1656)* APC(S1656[un/ph[.]], CK1e(S323[un][.]) T325[un][.]
→ T334[un][.] T337[un][.] S368[un][.] S405[un][.] T407[un][.]
→ S408[un][.]) | 1 APC1656-p @ 0.
→ {GeneralPhosphorylation}

KC1e*.().APC(T1655)* APC(S1652[ph[.]] T1655[un/ph[.]], CK1e(S323[un][.]) T325[un][.]
→ T334[un][.] T337[un][.] S368[un][.] S405[un][.]
→ T407[un][.] S408[un][.]) | 1 APC1655-p @ 0.
→ {GeneralPhosphorylation}

KC1e*.().APC(S1658)* APC(T1655[ph[.]] S1658[un/ph[.]], CK1e(S323[un][.]) T325[un][.]
→ T334[un][.] T337[un][.] S368[un][.] S405[un][.]
→ T407[un][.] S408[un][.]) | 1 APC1658-p @ 0.
→ {GeneralPhosphorylation}

KC1e*.().APC(T1661)* APC(S1658[ph[.]] T1661[un/ph[.]], CK1e(S323[un][.]) T325[un][.]
→ T334[un][.] T337[un][.] S368[un][.] S405[un][.]
→ T407[un][.] S408[un][.]) | 1 APC1661-p @ 0.
→ {GeneralPhosphorylation}

KC1e*.().APC(S1664)* APC(T1661[ph[.]] S1664[un/ph[.]], CK1e(S323[un][.]) T325[un][.]
→ T334[un][.] T337[un][.] S368[un][.] S405[un][.]
→ T407[un][.] S408[un][.]) | 1 APC1664-p @ 0.
→ {GeneralPhosphorylation}

// APC 20aaR5: S1857, D1860, S1861, S1863, D1866, D1969
KC1e*.().APC(S1861)* APC(S1861[un/ph[.]], CK1e(S323[un][.]) T325[un][.]
→ T334[un][.] T337[un][.] S368[un][.] S405[un][.] T407[un][.]
→ S408[un][.]) | 1 APC1861-p @ 0.
→ {GeneralPhosphorylation}

KC1e*.().APC(S1863)* APC(S1863[un/ph[.]], CK1e(S323[un][.]) T325[un][.]
→ T334[un][.] T337[un][.] S368[un][.] S405[un][.] T407[un][.]
→ S408[un][.]) | 1 APC1863-p @ 0.
→ {GeneralPhosphorylation}
A.1.2 Signalosome interactions

A.1.2.1 Interactions between Wnt, Fzd, and LRP

Wnt binds the NH2-terminal cystein rich extracellular domain of Fzd\textsuperscript{11}. Wnt also binds the extracellular domain of LRP\textsuperscript{81}, though this can be achieved through different \(\beta\)-propeller regions for different Wnt ligands\textsuperscript{31,13}. As we are not distinguishing between the Wnt ligands, we do not need to distinguish the sub-region location of the binding site. LRP and Fzd do not bind directly, but are associated in the presence of Wnt\textsuperscript{20}; we interpret this to mean a single Wnt can simultaneously bind both an LRP as well as a Fzd.

Fzd can dimerize, using its extracellular N-terminus\textsuperscript{16}. We do note that Fig3 in\textsuperscript{16} suggest higher-order oligomers than a mere dimer; however as the experiment relied on injecting multiple RNA constructs to induce higher expression, we believe it possible the higher-order polymers arose out of over-expression, and therefore opt for a more conservative assumption of mere dimerization.

LRP can dimerize using its C-terminus\textsuperscript{49}.

\[
\text{\texttt{'Fzld,Wnt+'}} \quad \text{Fzd(Fzd-dom[][1]), Wnt(Fzd[][1]) \@ ((1 / RescaleFactor) \ast \text{GeneralBinding}) \{((\text{RingClosureRate} \ast \text{GeneralBinding})}\}
\]

\[
\text{\texttt{'Fzld,Wnt-'}} \quad \text{Fzd(Fzd-dom[1][1]), Wnt(Fzd[1][1]) \@ GeneralUnbinding}
\]

\[
\text{\texttt{'LRP,LRP+'}} \quad \text{LRP(TermC[][1][1]), LRP(TermC[1][1]) \@ ((0.5 \ast (1 / RescaleFactor)) \ast \text{ReceptorDimerization}) \{((\text{RingClosureRate} \ast \text{ReceptorDimerization})}\}
\]
A.1.2.2 Interactions between CK1δ, CK1ε, and Dvl

CK1ε can bind and phosphorylate Dvl, whereas CK1δ is only shown to phosphorylate Dvl\(^{15}\). We find of interest two phosphorylation sites (or regions) on Dvl, both phosphorylated by either CK1δ and CK1ε.

The first site Serine 407, lies in proximity to the binding site used to bind Fzd. The second site is associated with Tyrosine 17\(^{94}\). We are aware CK1s are Serine/Threonine kinases, not Tyrosine kinases, and so they can’t phosphorylate Y17. We intend “Y17” to mean the region associated with said residue. This phosphorylation site (or region) lies within the DIX domain.
A.1.2.3 Co-polymerization of Axin and Dvl

Axin and Dvl can co-polymerize through concatenation of their respective DIX domains. This interaction is favored after treatment with CK1ε. With the realization CK1ε's activity is associated with a conserved residue on Dvl, we interpret the phosphorylation of the Y17 region increases the stability of the Dvl-Axin interaction. Thus the binding reaction observes the phosphorylation state of Dvl's Y17.

Dsh(DIX-head,Y17^),Axin(DIX-tail)+ \rightarrow Axin(DIX-tail[./1] Y17{un}[.]),  
\Rightarrow \{((1 / RescaleFactor) * GeneralBinding) / 1000\}

Dsh(DIX-tail,Y17^),Axin(DIX-head)+ \rightarrow Axin(DIX-head[./1] Y17{un}[.]),  
\Rightarrow \{((1 / RescaleFactor) * GeneralBinding) / 1000\}

Dsh(DIX-head,Y17^),Axin(DIX-tail)+ \rightarrow Axin(DIX-tail[./1] Y17{ph}[.]),  
\Rightarrow \{(RingClosureRate * GeneralBinding)\}

Dsh(DIX-tail,Y17^),Axin(DIX-head)+ \rightarrow Axin(DIX-head[./1] Y17{ph}[.]),  
\Rightarrow \{(RingClosureRate * GeneralBinding)\}
A.1.2.4 Binding of Fzd and Dvl

Fzd’s KTxxxW motif recruits Dvl’s PDZ domain\(^9\). The PDZ domain is in close proximity to the sites modified by CK1s. Additionally, their phosphorylation is associated with migration of the Dvl puncta from the cytosol to the plasma membrane. Moreover, a phospho-mimicking mutation induced membrane-bound Dvl complexes, while an un-phosphorylatable mutation prevents migration to the membrane\(^9\). We thus interpret the phosphorylation of S\(407\) increases affinity for the membrane-receptor Fzd.

\[ {\text{Fzd, Dsh(S407*)}} + \quad \text{Fzd(PDZ-bind[1.], Dvl(PDZ[1.], S407[ph]])} \ @ \ (1 / \text{RescaleFactor} / \text{GeneralBinding}) \{((\text{RingClosureRate} \ * \ \text{GeneralBinding}))} \]

\[ {\text{Fzd, Dsh(S407^*)}} + \quad \text{Fzd(PDZ-bind[1.], Dvl(PDZ[1.], S407[un]])} \ @ \ (((1 / \text{RescaleFactor}) / \text{GeneralBinding}) / 10000) \{((\text{RingClosureRate} \ * \ \text{GeneralBinding}) / 10000)} \]

\[ {\text{Fzd, Dsh(S407^-)}} \quad \text{Fzd(PDZ-bind[1.], Dvl(PDZ[1.], S407[un]])} \ @ \ \text{GeneralUnbinding} \]

A.1.2.5 Binding of Axin and LRP

Axin binds LRP, and this interaction competes with the binding of β-catenin\(^5\). This interaction is favored the more LRP is phosphorylated\(^6\). Assuming phosphorylation decreases the unbinding
rate, leaving the binding rate untouched, the ten phosphorylation sites yield $2^{10}$ unbinding rules. For brevity, we present the first and last two such rules.

1. \(\text{LRP,Axn(**********)}{}^{+}\)  
   \(\text{LRP}(S1490[^\text{ph}]\ldots T1530[^\text{ph}]\ldots T1572[^\text{ph}]\ldots)\)  
   \(\rightarrow\)  
   \(S1590[^\text{ph}]\ldots S1607[^\text{ph}]\ldots T1493[^\text{ph}]\ldots S1533[^\text{ph}]\ldots S1575[^\text{ph}]\ldots\)  
   \(\rightarrow\)  
   \(T1593[^\text{ph}]\ldots T1610[^\text{ph}]\ldots \text{PPPsP}[^0/\text{ph}], \text{Axin}(CBD[^0/\text{ph}]) @ ((1 / \text{RescaleFactor}) \times \text{GeneralBinding})\)  
   \((\text{RingClosureRate} \times \text{GeneralBinding})\)  

2. \(\text{LRP,Axn(**********)}{}^{-}\)  
   \(\text{LRP}(S1490[^\text{ph}]\ldots T1530[^\text{ph}]\ldots T1572[^\text{ph}]\ldots)\)  
   \(\rightarrow\)  
   \(S1590[^\text{ph}]\ldots S1607[^\text{ph}]\ldots T1493[^\text{ph}]\ldots S1533[^\text{ph}]\ldots S1575[^\text{ph}]\ldots\)  
   \(\rightarrow\)  
   \(T1593[^\text{ph}]\ldots T1610[^\text{ph}]\ldots \text{PPPsP}[^0/\text{ph}], \text{Axin}(CBD[^0/\text{ph}]) @ \text{GeneralUnbinding}\)  
   \((\text{GeneralUnbinding} \times 2)\)

A.1.2.6 Phosphorylation of LRP by CK1δ and GSK3β

The phosphorylation of LRP’s cytosolic terminus is carried out by GSK3β and a CK1δ. The phosphorylation sites are arranged into a 5-lobbed “rossette”, with two phosphorylation sites per lobe.
The first sites are primed by GSK3β. This satisfies the charge requirements of a CK1, which proceeds with phosphorylating the second sites in each lobe. The “rossette” fully assembled serves as the binding site for Axin.

Since LRP is a membrane receptor, it seems sensible to believe the CK1 family member mostly responsible for its phosphorylation would be CK1δ.

1
'GSK3B.().LRP(S1490)*' LRP(S1490{un/ph}[.]),
\[\text{GSK}()\]
\[\text{1 LRP-1490p @ 0.} \{\text{GeneralPhosphorylation}\}\]

2
'GSK3B.().LRP(T1530)*' LRP(T1530{un/ph}[.]),
\[\text{GSK}()\]
\[\text{1 LRP-1530p @ 0.} \{\text{GeneralPhosphorylation}\}\]

3
'GSK3B.().LRP(T1572)*' LRP(T1572{un/ph}[.]),
\[\text{GSK}()\]
\[\text{1 LRP-1572p @ 0.} \{\text{GeneralPhosphorylation}\}\]

4
'GSK3B.().LRP(S1590)*' LRP(S1590{un/ph}[.]),
\[\text{GSK}()\]
\[\text{1 LRP-1590p @ 0.} \{\text{GeneralPhosphorylation}\}\]

5
'GSK3B.().LRP(S1607)*' LRP(S1607{un/ph}[.]),
\[\text{GSK}()\]
\[\text{1 LRP-1607p @ 0.} \{\text{GeneralPhosphorylation}\}\]

6
'KC1δ*(.).LRP(T1493)*' LRP(S1490{ph}[.] T1493{un/ph}[.]),
\[\text{CK1d(S323{un}[.])]\]
\[\text{T325{un}[.] T334{un}[.] T337{un}[.] S368{un}[.] S405{un}[.]}\]
\[\text{T407{un}[.] S408{un}[.]}
\[\text{1 LRP-1493p @ 0.}\]
\[\{\text{GeneralPhosphorylation}\}\]

7
'KC1δ*(.).LRP(S1533)*' LRP(T1530{ph}[.] S1533{un/ph}[.]),
\[\text{CK1d(S323{un}[.])]\]
\[\text{T325{un}[.] T334{un}[.] T337{un}[.] S368{un}[.] S405{un}[.]}\]
\[\text{T407{un}[.] S408{un}[.]}
\[\text{1 LRP-1533p @ 0.}\]
\[\{\text{GeneralPhosphorylation}\}\]
A.1.3 Other interactions

A.1.3.1 Dvl polymerization

Dvl can polymerize through head-to-tail concatenation of its DIX domain. This interaction is disfavored after treatment with CK1ε. As mentioned for the Axin-Dvl interaction, the region associated with CK1ε modification lies within the DIX domain. We therefore interpret that modification of the Y17 region reduced the stability of the Dvl-Dvl interaction.
A.1.3.2 Activation of PP1, dephosphorylation of CKiδ and CKiε

In response to Wnt, Axin is dephosphorylated\(^8\). Moreover, the sites phosphorylated by CKiα are dephosphorylated by PPiA\(^62\). Likewise, in response to Wnt, both CKiδ and CKiε become active\(^15\), ergo their terminal tails dephosphorylated\(^69\), and this dephosphorylation can be carried out by PPiA, who binds their C-termini\(^18\). We assume this means PPiA becomes active under Wnt influence. As we do not know the mechanism, or what “active PPiA” means, we encode a flag with two states, which we use to test if PPiA is in “on” or “off” state.

One mechanism by which “activation” of PPiA could be accomplished is T316 dephosphorylation\(^33\) in response to Wnt. This would either by Wnt-mediated decrease of some kinase activity, or increase in some phosphatase activity. A phosphatase “activating” a phosphatase would not be a
new concept.

An alternative but more complicated mechanism would involve “Protein Phosphatase Inhibitor 2”. This inhibitor inhibits PP1A by binding it, so that the release of PP1A results in active phosphatase. There are two kinases known to destabilize this interaction through phosphorylation, GSK3β on Threonine 73 and ATM on Serine 44. Could it be that Wnt’s possible downstream effects through ATM (see section A.2) induce phosphorylation of Inhibitor 2, thereby activating PP1A?

''PP1A?' Wnt(), PP1(Act{off/on}[.]) | 1 PP1-a_on @ ((1 /
   → RescaleFactor) * GeneralBinding)
''PP1a;' PP1(Act{on/off}[.]) | 1 PP1-a_off @
   → GeneralUnbinding
''PP1A,KC1E+'' PP1(Recog[/.1]), CK1e(TermC[1/.]) @ ((1 / RescaleFactor) *
   → GeneralBinding) {(RingClosureRate * GeneralBinding)}
''PP1A,KC1E-' PP1(Recog[1/.]), CK1e(TermC[1/.]) @ GeneralUnbinding
''PP1a(.),CK1E(S323)^'' CK1e(S323{ph/un}[.]), PP1(Act{on}) | 1
   → CKe-323-u @ 0. {GeneralDephosphorylation}
''PP1a(.),CK1E(T325)^'' CK1e(T325{ph/un}[.]), PP1(Act{on}) | 1
   → CKe-325-u @ 0. {GeneralDephosphorylation}
''PP1a(.),CK1E(T334)^'' CK1e(T334{ph/un}[.]), PP1(Act{on}) | 1
   → CKe-334-u @ 0. {GeneralDephosphorylation}
''PP1a(.),CK1E(T337)^'' CK1e(T337{ph/un}[.]), PP1(Act{on}) | 1
   → CKe-337-u @ 0. {GeneralDephosphorylation}
''PP1a(.),CK1E(S368)^'' CK1e(S368{ph/un}[.]), PP1(Act{on}) | 1
   → CKe-368-u @ 0. {GeneralDephosphorylation}
''PP1a(.),CK1E(S405)^'' CK1e(S405{ph/un}[.]), PP1(Act{on}) | 1
   → CKe-405-u @ 0. {GeneralDephosphorylation}
''PP1a(.),CK1E(T407)^'' CK1e(T407{ph/un}[.]), PP1(Act{on}) | 1
   → CKe-407-u @ 0. {GeneralDephosphorylation}
''PP1a(.),CK1E(S408)^'' CK1e(S408{ph/un}[.]), PP1(Act{on}) | 1
   → CKe-408-u @ 0. {GeneralDephosphorylation}
A.1.3.3 PP1 regulation of PP2A

PP2A dephosphorylates many actors in the Wnt play, including Axin through B61[α,β,δ] \[^{85}\], APC through B61[α,δ] and/or B56α\[^{75,28}\], and β-catenin through B55α\[^{97}\]. Intriguingly, the core dimer of Cα and Aα may even dephosphorylate substrates without a recognition B subunit, as Aα can interact directly with Axin\[^{85}\]. We have difficulty reconciling the idea of a “constitutively active” phosphatase in the model. Unless we severely penalize the concentration of PP2A, reduce the rate constant of the rule, or somehow reduce the flow through the dephosphorylation reaction, we have
trouble observing the Wnt narrative emerge from the set of interactions, as PP2A maintains all the key players in an overwhelmingly dephosphorylated state.

Evidence of regulation points to PP1 somehow “activating” distinct heterotrimers of PP2A. It is unclear what this “activation” means mechanistically, but it seems sensible to believe PP1 may dephosphorylate some member of the PP2A heterotrimers. Would it dephosphorylate the B recognition subunit directly? Or perhaps act on the Aα and/or Cα particles? Given than the core Cα-Aα dimer may be able to directly interact with substrates, we assume the regulation is of the core dimer, and choose to reflect this as a fictitious state on the catalytic subunit Cα.

Moreover, could a heterotrimer dephosphorylate (i.e. “activate”) itself? This is a question of geometry, and as the heterotrimer appears largely an ordered structure, we do not believe it would be flexible enough to achieve an intra-complex dephosphorylation. Instead, as PP1 can bind the B55 and B56 subunits on their respective RxVxF and G/SILK/R+RxVxF motifs, we assume PP1 can “activate” directly the core dimer. This interpretation appears more consistent with the geometry of the proteins. Likewise, it is more conservative to conjure a single “activation” of the core dimer in two heterotrimers, rather than to conjure two distinct “activations” of two distinct recognition B subunits.

Our hypothesized mechanism would have “active” PP1 bind at B55’s RxVxF motif, “activating” its core dimer (e.g. through dephosphorylation) if present. An PP2-B55 heterotrimer loaded with an “active” core dimer would then be able to bind and dephosphorylate a B56 on S378. Once S378 is dephosphorylated, a PP1 could bind a B56 on its G/SILK/R+RxVxF motif, “activating” its core dimer if present. As all these activation reactions are balanced by unknown partners (e.g. kinases), we
invoke a 1st order de-activation rule to avoid unescapable outcomes (i.e. Escher Staircases), yielding a spontaneous phosphorylation for a phosphatase.

A.1.3.4 Dephosphorylation of β-catenin by PP2A

β-catenin residues S45, T41, S37, and S33, are dephosphorylated by protein phosphatase 2 A (PP2A), specifically using subunits Aα and Cα. However, this dephosphorylation is prevented by β-catenin being bound by APC, whereas Axin confers no such protection\(^79\). The recognition subunit involved in dephosphorylating β-catenin is B55α\(^97\). We thus interpret that PP2 can bind unbound β-catenin, as well as Axin-bound β-catenin, using the appropriate recognition subunit.

Can a heterotrimeric PP2A, loaded with a recognition subunit other than B55A, bound to Axin via subunit Aα\(^41\), dephosphorylate a β-catenin bound to that Axin, or a neighboring Axin? Does
the recognition B subunit confer only binding specificity, or also reaction specificity? Lacking more
insight into how the PP2A complex operates, specially with a scaffold protein like Axin bound, we
choose to represent the dephosphorylation reaction as requiring only the substrate and enzymatic
subunit co-located on the same complex.

```
1 'PP2.(.)Cat-b(S45)^' PP2_CA(Act{on} sub-a[.] sub-r[sub-c.PP2_B55A]),
→ bCat(S45{ph/un}[.]) | 1 Cat-S45-u @ 0.
→ {GeneralDephosphorylation}

2 'PP2.(.)Cat-b(T41)^' PP2_CA(Act{on} sub-a[.] sub-r[sub-c.PP2_B55A]),
→ bCat(T41{ph/un}[.]) | 1 Cat-T41-u @ 0.
→ {GeneralDephosphorylation}

3 'PP2.(.)Cat-b(S37)^' PP2_CA(Act{on} sub-a[.] sub-r[sub-c.PP2_B55A]),
→ bCat(S37{ph/un}[.]) | 1 Cat-S37-u @ 0.
→ {GeneralDephosphorylation}

4 'PP2.(.)Cat-b(S33)^' PP2_CA(Act{on} sub-a[.] sub-r[sub-c.PP2_B55A]),
→ bCat(S33{ph/un}[.]) | 1 Cat-S33-u @ 0.
→ {GeneralDephosphorylation}

5 'PP2,Cat-b+' PP2_B55A(Recog[.9[.]], bCat(iARM[.] PP2[.]) @ ((1 /
→ RescaleFactor) * GeneralBinding)

6 'PP2,Axin,Cat-b+' PP2_B55A(Recog[.9[.]], bCat(iARM[CBD.Axin] PP2[.]) @
→ ((1 / RescaleFactor) * GeneralBinding)

7 'PP2,Cat-b-' PP2_B55A(Recog[9[.]], bCat(PP2[9[.]]) @ GeneralUnbinding
```

A.1.3.5 Dephosphorylation of APC by PP2A

APC can use a site proximal to its ARM domain to bind the recognition B subunit of PP2A, B56α.28
Knowing that the 20 amino-acid repeats are phosphorylated, and assuming they are dephospho-
rylatable, we assume PP2 can dephosphorylate APC’s repeats.
// APC 20aaR1: S1272, S1275, S1276, S1278, S1281, E1284

'PP2-B56a().APC(S1276)^' APC(S1276{ph/un}[.]), PP2_CA(Act[on] sub[a[_.

↓ sub[r[sub-c.PP2_B56A]]) 1 APC1276-u @ 0.

↓ {GeneralDephosphorylation}

'PP2-B56a().APC(S1272)^' APC(S1272{ph/un}[.]), PP2_CA(Act[on] sub[a[_.

↓ sub[r[sub-c.PP2_B56A]]) 1 APC1272-u @ 0.

↓ {GeneralDephosphorylation}

'PP2-B56a().APC(S1275)^' APC(S1275{ph/un}[.]), PP2_CA(Act[on] sub[a[_.

↓ sub[r[sub-c.PP2_B56A]]) 1 APC1275-u @ 0.

↓ {GeneralDephosphorylation}

'PP2-B56a().APC(S1278)^' APC(S1278{ph/un}[.]), PP2_CA(Act[on] sub[a[_.

↓ sub[r[sub-c.PP2_B56A]]) 1 APC1278-u @ 0.

↓ {GeneralDephosphorylation}

// APC 20aaR2: S1386, T1389, S1390, S1392, D1395, E1398

'PP2-B56a().APC(S1390)^' APC(S1390{ph/un}[.]), PP2_CA(Act[on] sub[a[_.

↓ sub[r[sub-c.PP2_B56A]]) 1 APC1390-u @ 0.

↓ {GeneralDephosphorylation}

'PP2-B56a().APC(S1386)^' APC(S1386{ph/un}[.]), PP2_CA(Act[on] sub[a[_.

↓ sub[r[sub-c.PP2_B56A]]) 1 APC1386-u @ 0.

↓ {GeneralDephosphorylation}

'PP2-B56a().APC(T1389)^' APC(T1389{ph/un}[.]), PP2_CA(Act[on] sub[a[_.

↓ sub[r[sub-c.PP2_B56A]]) 1 APC1389-u @ 0.

↓ {GeneralDephosphorylation}

'PP2-B56a().APC(S1392)^' APC(S1392{ph/un}[.]), PP2_CA(Act[on] sub[a[_.

↓ sub[r[sub-c.PP2_B56A]]) 1 APC1392-u @ 0.

↓ {GeneralDephosphorylation}

// APC 20aaR3: S1501, S1504, S1505, S1507, S1510, E1513

'PP2-B56a().APC(S1505)^' APC(S1505{ph/un}[.]), PP2_CA(Act[on] sub[a[_.

↓ sub[r[sub-c.PP2_B56A]]) 1 APC1505-u @ 0.

↓ {GeneralDephosphorylation}
//'PP2-B56a.().APC(S1501)^' APC(S1501{ph/un}[.]), PP2_CA(Act{on} sub-a[[_])
  → sub-r[sub-c.PP2_B56A])  | 1 APC1501-u @ 0.
  → {GeneralDephosphorylation}

'PP2-B56a.().APC(S1504)^' APC(S1504{ph/un}[.]), PP2_CA(Act{on} sub-a[[_)
  → sub-r[sub-c.PP2_B56A])  | 1 APC1504-u @ 0.
  → {GeneralDephosphorylation}

'PP2-B56a.().APC(S1507)^' APC(S1507{ph/un}[.]), PP2_CA(Act{on} sub-a[[_)
  → sub-r[sub-c.PP2_B56A])  | 1 APC1507-u @ 0.
  → {GeneralDephosphorylation}

'PP2-B56a.().APC(S1510)^' APC(S1510{ph/un}[.]), PP2_CA(Act{on} sub-a[[_)
  → sub-r[sub-c.PP2_B56A])  | 1 APC1510-u @ 0.
  → {GeneralDephosphorylation}

// APC 20aaR4: S1652, T1655, S1656, S1658, T1661, S1664

'PP2-B56a.().APC(S1656)^' APC(S1656{ph/un}[.]), PP2_CA(Act{on} sub-a[[_)
  → sub-r[sub-c.PP2_B56A])  | 1 APC1656-u @ 0.
  → {GeneralDephosphorylation}

'PP2-B56a.().APC(S1652)^' APC(S1652{ph/un}[.]), PP2_CA(Act{on} sub-a[[_)
  → sub-r[sub-c.PP2_B56A])  | 1 APC1652-u @ 0.
  → {GeneralDephosphorylation}

'PP2-B56a.().APC(T1655)^' APC(T1655{ph/un}[.]), PP2_CA(Act{on} sub-a[[_)
  → sub-r[sub-c.PP2_B56A])  | 1 APC1655-u @ 0.
  → {GeneralDephosphorylation}

'PP2-B56a.().APC(S1658)^' APC(S1658{ph/un}[.]), PP2_CA(Act{on} sub-a[[_)
  → sub-r[sub-c.PP2_B56A])  | 1 APC1658-u @ 0.
  → {GeneralDephosphorylation}

'PP2-B56a.().APC(T1661)^' APC(T1661{ph/un}[.]), PP2_CA(Act{on} sub-a[[_)
  → sub-r[sub-c.PP2_B56A])  | 1 APC1661-u @ 0.
  → {GeneralDephosphorylation}

'PP2-B56a.().APC(S1664)^' APC(S1664{ph/un}[.]), PP2_CA(Act{on} sub-a[[_)
  → sub-r[sub-c.PP2_B56A])  | 1 APC1664-u @ 0.
  → {GeneralDephosphorylation}

// APC 20aaR5: S1857, D1860, S1861, S1863, D1866, D1969
26 'PP2-B56a.(.)APC(S1861)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC1861-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]

27 'PP2-B56a.(.)APC(S1857)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC1857-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]

28 'PP2-B56a.(.)APC(S1863)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC1863-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]


30 'PP2-B56a.(.)APC(S1968)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC1968-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]

31 'PP2-B56a.(.)APC(S1964)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC1964-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]

32 'PP2-B56a.(.)APC(S1967)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC1967-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]

33 'PP2-B56a.(.)APC(S1970)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC1970-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]

34 'PP2-B56a.(.)APC(S1973)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC1973-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]

35 // APC 20aaR7: S2022, S2025, S2026, S2028, S2031, S2034

36 'PP2-B56a.(.)APC(S2026)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC2026-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]

37 'PP2-B56a.(.)APC(S2022)\(^{ph/un}\).PP2_CA(Act(on) sub-a[_.])
\[\text{1 APC2022-u @ 0.}\]
\[\{\text{GeneralDephosphorylation}\}\]
A.1.3.6 Assembly of the PP2A heterotrimeric complex

From PDB crystal structure 4I5L, PP2A is a heterotrimeric complex. Specifically, the interactions between members are rigid, and do not allow higher order complexes to assemble. Thus the binding of one protomer to another depends on the bound state of both. The ruleset to assemble the trimer therefore observes the interfaces ancillary to the binding event; a monomer binds a monomer to create a dimer, a dimer binds a monomer to create a trimer; this thusly disallows the binding of two dimers, which would have resulted in a tetramer.
'PP2|B56a_PP2|Alfa+/-' PP2_B56A(sub-a[1] sub-c[.]), PP2_AA(sub-c[.)
⇒ sub-r[1/.)] @ ((1 / RescaleFactor) * GeneralBinding)

'PP2|B56a_PP2|Alfa+/-_op' PP2_B56A(sub-a[1/] sub-c[.]), PP2_AA(sub-c[.)
⇒ sub-r[1/.]) @ GeneralUnbinding

'PP2|Cata_PP2|Alfa+/-' PP2_CA(sub-a[1] sub-r[.]), PP2_AA(sub-c[.)/1]
⇒ sub-r[.] @ ((1 / RescaleFactor) * GeneralBinding)

'PP2|Cata_PP2|Alfa+/-_op' PP2_CA(sub-a[1/] sub-r[.]), PP2_AA(sub-c[.)/1]
⇒ sub-r[.] @ GeneralUnbinding

'PP2|Cata_PP2|Alfa_PP2|B56a+/-' PP2_CA(sub-a[1] sub-r[./2]),
⇒ PP2_AA(sub-c[1] sub-r[./3]), PP2_B56A(sub-a[./3] sub-c[./2]) @ ((1 /
⇒ RescaleFactor) * GeneralBinding)

'PP2|Cata_PP2|Alfa_PP2|B56a+/-_op' PP2_CA(sub-a[1] sub-r[./2]),
⇒ PP2_AA(sub-c[1] sub-r[./3]), PP2_B56A(sub-a[./3] sub-c[./2]) @
⇒ ((EntropyFactor * GeneralUnbinding) * GeneralUnbinding)

'PP2|B56a_PP2|Cata+/-' PP2_B56A(sub-a[1] sub-c[.2]),
⇒ PP2_AA(sub-c[.3] sub-r[1]), PP2_CA(sub-a[./3] sub-r[./2]) @ ((1 /
⇒ RescaleFactor) * GeneralBinding)

'PP2|B56a_PP2|Cata+/-_op' PP2_B56A(sub-a[1] sub-c[.2]),
⇒ PP2_AA(sub-c[.3] sub-r[1]), PP2_CA(sub-a[./3] sub-r[./2]) @
⇒ ((EntropyFactor * GeneralUnbinding) * GeneralUnbinding)

'PP2|B56a_PP2|Cata+/-' PP2_B56A(sub-a[.2] sub-c[1]),
⇒ PP2_CA(sub-a[./3] sub-r[1]), PP2_AA(sub-c[./3] sub-r[./2]) @ ((1 /
⇒ RescaleFactor) * GeneralBinding)

'PP2|B56a_PP2|Cata+/-_op' PP2_B56A(sub-a[./2] sub-c[1]),
⇒ PP2_CA(sub-a[.3] sub-r[1]), PP2_AA(sub-c[.3] sub-r[.2]) @
⇒ ((EntropyFactor * GeneralUnbinding) * GeneralUnbinding)

'PP2|B55a_PP2|Cata+/-' PP2_B55A(sub-a[.] sub-c[.1]), PP2_CA(sub-a[.]
⇒ sub-r[.1]) @ ((1 / RescaleFactor) * GeneralBinding)

'PP2|B55a_PP2|Cata+/-_op' PP2_B55A(sub-a[.1] sub-c[.]), PP2_CA(sub-a[.]
⇒ sub-r[.1]) @ GeneralUnbinding

'PP2|B55a_PP2|Alfa+/-' PP2_B55A(sub-a[1/.) sub-c[.]), PP2_AA(sub-c[.)
⇒ sub-r[1/.)] @ ((1 / RescaleFactor) * GeneralBinding)

'PP2|B55a_PP2|Alfa+/-_op' PP2_B55A(sub-a[1/.) sub-c[.]), PP2_AA(sub-c[.)
⇒ sub-r[1/.)] @ GeneralUnbinding
A.1.3.7 Recruitment and dephosphorylation of Axin by PP1

PP1 can bind Axin towards the C-terminus, and dephosphorylate residues S80, S82, S222, and S473. It is unknown to us what phosphatase (if any) is capable of dephosphorylating the rest of the GSK3β or CK1α modified sites, T481, S486, and S493, all lying within the β-catenin binding site. We assume a plausible phosphatase is PP1, as it already binds and dephosphorylates other Axin sites. As the dephosphorylation of Axin requires Wnt, we introduce the requirement PP1 be “active”.

An alternative to PP1 would be PP2A, as the regulatory A subunit α(Aα) binds Axin directly;
and that would open the possibility of having two separate regulations on Axin: PP1 mediating
stability, PP2A mediating β-catenin affinity. Wanting more evidence for the latter, specifically in
terms of which (if any) regulatory B and C units of PP2 would be involved in assembling the het-
erotrimeric PP2A complex, we choose the more conservative former and assume PP1 is capable of
dephosphorylating all these sites.

A.1.3.8 Autoinhibition of CK1δ and CK1ε

Both CK1δ and CK1ε posses C-terminal tails that can be auto-phosphorylated intra-molecularly,
and act as auto-inhibitory when phosphorylated.69
'CK1E(S323)*': CK1e(S323{un/ph}[]) | 1 CKe-323-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1E(T325)*': CK1e(T325{un/ph}[]) | 1 CKe-325-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1E(T334)*': CK1e(T334{un/ph}[]) | 1 CKe-334-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1E(T337)*': CK1e(T337{un/ph}[]) | 1 CKe-337-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1E(S368)*': CK1e(S368{un/ph}[]) | 1 CKe-368-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1E(S405)*': CK1e(S405{un/ph}[]) | 1 CKe-405-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1E(T407)*': CK1e(T407{un/ph}[]) | 1 CKe-407-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1D(S323)*': CK1d(S323{un/ph}[]) | 1 CKd-323-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1D(T325)*': CK1d(T325{un/ph}[]) | 1 CKd-325-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1D(T334)*': CK1d(T334{un/ph}[]) | 1 CKd-334-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1D(T337)*': CK1d(T337{un/ph}[]) | 1 CKd-337-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1D(S368)*': CK1d(S368{un/ph}[]) | 1 CKd-368-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1D(S405)*': CK1d(S405{un/ph}[]) | 1 CKd-405-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1D(T407)*': CK1d(T407{un/ph}[]) | 1 CKd-407-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]

'CK1D(S408)*': CK1d(S408{un/ph}[]) | 1 CKd-408-p @
\[\xrightarrow{\text{GeneralPhosphorylation}}\]
A.13.9 LRP dephosphorylation

This ruleset is present simply to avoid “Escher staircases”, i.e. states of agents that are irreversible or unavoidable. LRP is phosphorylated, ergo it must be dephosphorylatable. Lacking a known enzyme for this task, we represent a spontaneous auto-dephosphorylation for each site, occurring at rate slower than phosphatase-catalyzed events.

1 'LRP(S1490)' → LRP(S1490)[ph/un][.] | 1 LRP1490-u @ (GeneralDephosphorylation / 100)
2 'LRP(T1530)' → LRP(T1530)[ph/un][.] | 1 LRP1530-u @ (GeneralDephosphorylation / 100)
3 'LRP(T1572)' → LRP(T1572)[ph/un][.] | 1 LRP1572-u @ (GeneralDephosphorylation / 100)
4 'LRP(S1590)' → LRP(S1590)[ph/un][.] | 1 LRP1590-u @ (GeneralDephosphorylation / 100)
5 'LRP(S1607)' → LRP(S1607)[ph/un][.] | 1 LRP1607-u @ (GeneralDephosphorylation / 100)
6 'LRP(T1493)' → LRP(T1493)[ph/un][.] | 1 LRP1493-u @ (GeneralDephosphorylation / 100)
7 'LRP(S1533)' → LRP(S1533)[ph/un][.] | 1 LRP1533-u @ (GeneralDephosphorylation / 100)
8 'LRP(S1575)' → LRP(S1575)[ph/un][.] | 1 LRP1575-u @ (GeneralDephosphorylation / 100)
9 'LRP(T1593)' → LRP(T1593)[ph/un][.] | 1 LRP1593-u @ (GeneralDephosphorylation / 100)
10 'LRP(T1610)' → LRP(T1610)[ph/un][.] | 1 LRP1610-u @ (GeneralDephosphorylation / 100)
A.2 Non-canonical Wnt

These interactions are not part of the Wnt model. They were found during the literature search, and are included here as a resource for other modelers.

A.2.0.1 Non-canonical interactions

This block contains several elements common to the $Ca^{2+}$ “non-cannonical” Wnt pathway. They are of relevance because they affect the transcriptional activity of the LEF1-β-catenin complex while taking as input Wnti $^{77}$. This block makes one assumption to fill the gap between Wnt input and MAP3K activation, and that hole is filled in by ATM and TRAF6.

The mode of action of this pathway is complicated, but upon induced DNA-damage-related Ca2+ influx, ATM translocates to the cytosol $^{39}$. We speculate that Wnt induced Ca2+ influx can also translocate ATM. Once in the cytosol, ATM can oligomerize, usually into 2mer or 4mer conformations. This oligomerizations bind the 1961-2046 region (Oligo) of one monomer, to a catalytic region of the next $^{7}$. Of special interest is the fact that ATM con bind TRAF6 through the 2152-2157 region $^{39}$, which happens to be VERY close to Oligo. Considering that TRAF6 can also multimerize $^{37}$, it is logical to assume that ATM polymerization induces TRAF6 aggregation. Another key step in this cascade is the activation of the TRAF6 ubiquitination activity upon multimerization, and subsequent auto-ubiquitination $^{37}$. This phenomenon leads to two events: firstly active TRAF6 binds and ubiquitinates MAP3K7 $^{86}$. Secondly, ubiquitinated TRAF6 is recognizable by TAB2’z zinc finger domain $^{37}$. As this three components are found together $^{86}$, it is tempting to postulate that
TRAF6 acts as a scaffold to assemble MAP3K7 to TAB1. Once MAP3K7 and TAB1 are assembled, MAP3K7 autophosphorylates and activates\(^7\). Active MAP3K7 then phosphorylates NLK, which activates it and triggers its translocation to the nucleus\(^7\). Once in the nucleus, it phosphorylates LEF1/TCF. The ensuring complex formation and nuclear export of TCFs mediated by 14-3-3 lies well beyond the scope of our modeling efforts, as it heavily relies on compartment shuttling.

\[
\begin{align*}
\text{'ATM.ATM+'} & \quad \text{ATM}[^{\text{Dim}}].[^{1/1}], \text{ATM}[^{\text{Catalytic}}].[^{1/1}] @ (1/ \text{'RescaleFactor'}) * \\
\quad & \quad \quad \quad \rightarrow \, \text{'GeneralBinding'} \{ \text{'RingClosureRate'} * \text{'GeneralBinding'} \} \\
\text{'ATM.ATM-'} & \quad \text{ATM}[^{\text{Dim}}].[1/1], \text{ATM}[^{\text{Catalytic}}][1/1] \quad @ \text{'GeneralUnbinding'} \\
\text{'ATM.TRAF6+'} & \quad \text{ATM}[^{\text{TBM}}].[^{1/1}], \text{ATM}[^{\text{MATH}}_a].[^{1/1}] \quad @ (1/ \text{'RescaleFactor'}) * \\
\quad & \quad \quad \quad \rightarrow \, \text{'GeneralBinding'} \{ \text{'RingClosureRate'} * \text{'GeneralBinding'} \} \\
\text{'ATM.TRAF6-'} & \quad \text{ATM}[^{\text{TBM}}][1/1], \text{ATM}[^{\text{MATH}}_a][1/1] \quad @ \text{'GeneralUnbinding'} \\
\text{'TRAF6.TRAF6+'} & \quad \text{ATM}[^{\text{MATH}}_b].[1/1], \text{TRAF6}[^{\text{MATH}}_b][1/1] \quad @ 0.5 * (1/ \\
\quad & \quad \quad \quad \rightarrow \, \text{'RescaleFactor'}) * \text{'GeneralBinding'} \{ \text{'RingClosureRate'} * \\
\quad & \quad \quad \quad \rightarrow \, \text{'GeneralUnbinding'} \} \quad \quad \text{// Automorphism preserved, rate halved} \\
\text{'TRAF6.TRAF6-'} & \quad \text{ATM}[^{\text{MATH}}_b][1/1], \text{TRAF6}[^{\text{MATH}}_b][1/1] \quad @ 0.5 * \\
\quad & \quad \quad \quad \rightarrow \, \text{'RescaleFactor'}) * \text{'GeneralUnbinding'} \{ \text{'RingClosureRate'} * \\
\quad & \quad \quad \quad \rightarrow \, \text{'GeneralUnbinding'} \} \quad \quad \text{// Automorphism preserved, rate halved} \\
\text{'TRAF6.MAP3K7+'} & \quad \text{TRAF6}[^{\text{MATH}}_b][\text{Lys63}^{\text{ub}}][1/1], \text{M3K7}[^{\text{Lys63}}^{\text{ub}}][1/1] \quad @ (1/ \\
\quad & \quad \quad \quad \rightarrow \, \text{'RescaleFactor'}) * \text{'GeneralBinding'} \{ \text{'RingClosureRate'} * \\
\quad & \quad \quad \quad \rightarrow \, \text{'GeneralUnbinding'} \} \\
\text{'TRAF6.MAP3K7-'} & \quad \text{TRAF6}[^{\text{TRAFF}}][1/1], \text{M3K7}[^{\text{Lys63}}^{\text{un}}][1/1] \quad @ \\
\quad & \quad \quad \quad \rightarrow \, \text{'RescaleFactor'}) * \text{'GeneralUnbinding'} \\
\text{'TRAF6.MAP3K7@'} & \quad \text{TRAF6}[^{\text{MATH}}_b][\text{Lys63}^{\text{ub}}][9], \text{M3K7}[^{\text{Lys63}}^{\text{ub}}][9] \quad @ \\
\quad & \quad \quad \quad \rightarrow \, \text{'RescaleFactor'}) * \text{'GeneralUnbinding'} \\
\text{'(TRAF6@).TAB2+'} & \quad \text{TRAF6}[^{\text{Lys63}^{\text{ph}}}[1/1], \text{TAB2}[^{\text{NZF}}][1/1] \quad @ (1/ \\
\quad & \quad \quad \quad \rightarrow \, \text{'RescaleFactor'}) * \text{'GeneralUnbinding'} \\
\text{'(TRAF6@).TAB2-'} & \quad \text{TRAF6}[^{\text{Lys63}^{\text{ub}}}[1/1], \text{TAB2}[^{\text{NZF}}][1/1] \quad @ \text{'GeneralUnbinding'} \\
\text{'TAB2.MAP3K7+'} & \quad \text{TRAF6}[^{\text{i_MAP3K7}}][1/1], \text{M3K7}[^{\text{i TAB2}}][1/1] \quad @ (1/ \\
\quad & \quad \quad \quad \rightarrow \, \text{'RescaleFactor'}) * \text{'GeneralUnbinding'} \\
\text{'TAB2.MAP3K7-'} & \quad \text{TRAF6}[^{\text{i_MAP3K7}}][1/1], \text{M3K7}[^{\text{i TAB2}}][1/1] \quad @ \text{'GeneralUnbinding'} \\
\text{'MAP3K7|Thr184*'} & \quad \text{M3K7}[^{\text{i TAB2}}][\text{Thr184}^{\text{ph}}], \text{Thr184}^{\text{un/ph}} \quad @ \text{'GeneralPhosphorylation'}
\end{align*}
\]
A.2.0.2 Transcriptional feedbacks

These are various feedbacks on the canonical pathway that rely on proteins being transcribed, mostly as a result of β-catenin accumulation and translocation to the nucleus. As our modeling effort concerns a time-scale shorter than what would be required to observe the transcriptional feedbacks, we do not include these in the main model. These feedbacks include Axin2 (a.k.a. Conductin)\textsuperscript{46}, Naked-cuticle\textsuperscript{95}, and Dickkopf\textsuperscript{32}.
The impact of Conductin seems straightforward, as it is claimed to be physiologically identical to Axin. Production of Conductin could therefore simply increase the concentration of the agent Axin.

Dkk is a secreted protein that binds LPR. It is mentioned through the action of the Kremen receptor, Dkk induces the endocytosis of LRP. However, since Kremen’s presence appears to only play a role when LRP is over-expressed, we only consider the competitive binding of Dkk to LRP. Endocytosis of LRP could very well be carried out through the endocytosis of the signalosome.

Through PR72, a regulatory subunit of PP2, Nkd is able to bind and recruit Dvl into an phosphatase-inactive complex. We are assuming the mechanism is through direct occupation of the recognition site on PP2, while Nkd retains the ability to bind Dvl. It may also destabilize Dsh, possibly through CYLD mediated deubiquitination, though that is less clear.

1. '/Nkd/' bCat(iARM[9]), LEF1(CBD[9], HMGbox[1]), Genome(i_BS[1]), NKD1() @ 0.1
2. '/Dkk/' bCat(iARM[9]), LEF1(CBD[9], HMGbox[1]), Genome(i_BS[1]), DKK1() @ 0.1
3. '/Axin2/' bCat(iARM[9]), LEF1(CBD[9], HMGbox[1]), Genome(i_BS[1]), Axin() @ 0.1
4. 'Dkk.LPR+' DKK1(CRD_2[.]/1), LRP(iWnt[.]/1) @ (1/ 'RescaleFactor' ) * 'GeneralBinding'
5. 'Dkk.LPR-' DKK1(CRD_2!1), LRP(iWnt!1) -> DKK1(CRD_2), LRP(iWnt) @ 'GeneralUnbinding'
6. 'PP2|P72.Nkd+' PP2_B72(iRecog[.]/1), NKD1(i_PP2[.]/1) @ (1/ 'RescaleFactor' ) * 'GeneralBinding' { 'RingClosureRate' * 'GeneralBinding' }
7. 'PP2|P72.Nkd-' PP2_B72(iRecog!1), NKD1(i_PP2!1) -> PP2_B72(iRecog), NKD1(i_PP2) @ 'GeneralUnbinding'
These serve to remind the author about basic maths and are not presented as contributions of merit.

B.1  The two body problem

This is the derivation I use to characterize the bond between two agent types, knowing their abundances (in molecules) and the dissociation constant (in molecules), once the system reaches ther-
modynamic equilibrium. This is a characterization of a stochastic system, and therefore does not assume any given volume or concentrations.

Some definitions:

$A_t$ total amount of agents of type A

$A_f$ amount of free agents of type A; i.e. A monomers

$B_t$ total amount of agents of type B

$B_f$ amount of free agents of type B; i.e. B monomers

$AB$ total amount of dimers containing one A and one B

$\Gamma_D$ the stochastic dissociation constant for the A-B bond; ratio of the unbinding rate by the binding rate for the interaction, analogous to the deterministic dissociation constant $K_D$
The derivation:

\[ \Gamma_D = \frac{A_B f}{AB} \]

with

\[ B_t = AB + B_f \]

\[ \therefore \quad \Gamma_D = \frac{A(B_t - AB)}{AB} \]

\[ \Leftrightarrow \quad \Gamma_D AB = B_t A_f - ABA_f \]

\[ \Leftrightarrow \quad AB(\Gamma_D + A_f) = B_t A_f \]

\[ \Leftrightarrow \quad \frac{AB}{\Gamma_D} = \frac{A_f}{\Gamma_D + A_f} \]

with

\[ A_t = AB + A_f \]

\[ \therefore \quad \frac{AB}{\Gamma_D} = \frac{A_t - AB}{A_t - AB + \Gamma_D} \]

\[ \Leftrightarrow \quad AB = B_t \frac{A_t - AB}{A_t - AB + \Gamma_D} \]

\[ \Leftrightarrow \quad AB(A_t - AB + \Gamma_D) = B_t(A_t - AB) \]

\[ \Leftrightarrow \quad A_t AB - AB^2 + \Gamma_D AB = A_t B_t - B_t AB \]

\[ \Leftrightarrow \quad AB(A_t + B_t + \Gamma_D) - AB^2 = A_t B_t \]

\[ \Leftrightarrow \quad -AB^2 + AB(A_t + B_t + \Gamma_D) - A_t B_t = 0 \]

\[ \therefore \quad AB = \frac{-(A_t + B_t + \Gamma_D) \pm \sqrt{(A_t + B_t + \Gamma_D)^2 - 4(A_t B_t)}}{2} \]

The quadratic equation yields a discriminant \( \Delta = (A_t + B_t + \Gamma_D)^2 - 4A_t B_t \). The quadratic component of the discriminant increases quadratically with \( A_t, B_t, \) and \( \Gamma_D \), whereas the linear component decreases linearly with \( A_t \) and \( B_t \). As the abundances and dissociation constant must be greater than 1 (for biologically significant systems), \( \forall A_t, B_t, \Gamma_D, \Delta > 0 \). Of the two solutions, taking the positive solution would break mass conservation, therefore I take the negative solution.

256
\[ AB = \frac{1}{2} (A_i + B_i + \Gamma_D - \sqrt{(A_i + B_i + \Gamma_D)^2 - 4A_i B_i}) \] (B.1)
Definition of the patterns used as observables

Here I present the definition for the patterns reported by the Kappa simulator as a function of time, nicknamed “observables”. My figure producing pipeline uses the name of the observable in the plot, and so the names used throughout this work can be looked up here. An observable declaration is of
the form of a label, followed by pipe symbols enclosing a Kappa pattern. The label is the observable name printed on plots.

```
%obs: '[Wnt]' | Wnt()
%obs: '[Wnt.LRP]' | Wnt(LRP[1]), LRP(Wnt[1])
%obs: '[Wnt.Fzd]' | Wnt(Fzd[1]), Fzd(Fzd-dom[1])
%obs: '[Wnt.LRP.Fzd]' | Wnt(Fzd[1] LRP[2]), LRP(Wnt[2]), Fzd(Fzd-dom[1])
%obs: '[Fzd.CRD]' | Fzd(CRD[1]), Fzd(CRD[1])
%obs: '[LRP.LRP]' | LRP(TermC[1]), LRP(TermC[1])
%obs: '[Fzd.Dvl]' | Fzd(PDZ-bind[1]), Dvl(PDZ[1])
%obs: '[Dsh3|Y17-ph]' | Dvl(Y17{ph})
%obs: '[Dsh3|Y17-un]' | Dvl(Y17{un})
%obs: '[Dsh3|S407-ph]' | Dvl(S407{ph})
%obs: '[Dsh3|S407-un]' | Dvl(S407{un})
%obs: '[Dsh3-overall]' | Dvl()
%obs: '[Dsh3-b.Axn-a]' | Dvl(DIX-tail[1]), Axin(DIX-head[1])
%obs: '[Dsh3-a.Axn-b]' | Dvl(DIX-head[1]), Axin(DIX-tail[1])
%obs: '[PP1-on]' | PP1(Act{on}[#])
%obs: '[PP1-off]' | PP1(Act{off}[#])
%obs: '[KC1E]' | CK1e()
%obs: '[KC1E|S1-8_u_n]' | CK1e(S323{un} T325{un} T334{un} T337{un} S368{un} ← S405{un} T407{un} S408{un})
%obs: '[KC1D]' | CK1d()
%obs: '[KC1D|S1-8_u_n]' | CK1d(S323{un} T325{un} T334{un} T337{un} S368{un} ← S405{un} T407{un} S408{un})
%obs: '[Axn.LRP]' | LRP(PPPsP[1]), Axin(CBD[1])
%obs: '[Axn.CatB]' | Axin(CBD[1]), bCat(iARM[1])
%obs: '[LRP|S1490_p]' | LRP(S1490{ph})
%obs: '[LRP|T1493_p]' | LRP(T1493{ph})
%obs: '[LRP_h_y_p_e_r_p_h]' | LRP(S1490{ph} T1530{ph} T1572{ph} S1590{ph} ← S1607{ph} T1493{ph} S1533{ph} S1575{ph} T1593{ph} T1610{ph})
%obs: '[Axn.CK1a]' | Axin(PP1[1]), CK1a(CatDom[1])
%obs: '[Axn.CK1d]' | Axin(PP1[1]), CK1d(CatDom[1])
```
%obs: '[Axn.CK1e]' |Axin(PP1[1]), CK1e(CatDom[1])
%obs: '[Axn.PP1]' |Axin(PP1[1]), PP1(Recog[1])
%obs: '[Axn.PP1_e_m_p_t_y]' |Axin(PP1[.])
%obs: '[Axn.GSK]' |Axin(GSK[1]), GSK(Axin[1])
%obs: '[CK1a.Axn.GSK]' |Axin(GSK[1] PP1[2]), CK1a(CatDom[2]), → GSK(Axin[1])
%obs: '[Axin]' |Axin()
%obs: '[Axin.Axin]' |Axin(DIX-head[1]), Axin(DIX-tail[1])
%obs: '[Axin|CatBD_e_m_p_t_y]' |Axin(CBD[.])
%obs: '[Axin|CatBD_h_y_p_e_r_p_h]' |Axin(S469{ph} T481{ph} S493{ph})
%obs: '[Axin.Proteasome]' |Axin(Prot[1]), Prot(PSMD2[1])
%obs: '[Axin1.bTrCP]' |Axin(bTrCP[1]), bTrCP(Axin[1])
%obs: '[APC|SAMP1.Axin]' |Axin(RGS[1]), APC(SAMP-1[1])
%obs: '[APC|SAMP2.Axin]' |Axin(RGS[1]), APC(SAMP-2[1])
%obs: '[APC|SAMP3.Axin]' |Axin(RGS[1]), APC(SAMP-3[1])
%obs: '[APC]' |APC()
%obs: '[APC.APC]' |APC(Oligo[1]), APC(Oligo[1])
%obs: '[APC|rpt20AA1_e_m_p_t_y]' |APC(rpt20AA1[.])
%obs: '[APC|rpt20AA2_e_m_p_t_y]' |APC(rpt20AA2[.])
%obs: '[APC|rpt20AA3_e_m_p_t_y]' |APC(rpt20AA3[.])
%obs: '[APC|rpt20AA4_e_m_p_t_y]' |APC(rpt20AA4[.])
%obs: '[APC|rpt20AA5_e_m_p_t_y]' |APC(rpt20AA5[.])
%obs: '[APC|rpt20AA6_e_m_p_t_y]' |APC(rpt20AA6[.])
%obs: '[APC|rpt20AA7_e_m_p_t_y]' |APC(rpt20AA7[.])
%obs: '[APC|rpt20AA_1_-_7_e_m_p_t_y]' (((((('[APC|rpt20AA1_e_m_p_t_y]' + → '[APC|rpt20AA2_e_m_p_t_y]' + '[APC|rpt20AA3_e_m_p_t_y]') + → '[APC|rpt20AA4_e_m_p_t_y]') + '[APC|rpt20AA5_e_m_p_t_y]') + → '[APC|rpt20AA6_e_m_p_t_y]') + '[APC|rpt20AA7_e_m_p_t_y]')
%obs: '[APC|rpt20AA1_h_y_p_e_r_p_h]' |APC(S1272{ph} S1275{ph} S1276{ph} S1278{ph} S1281{ph})
%obs: '[APC|rpt20AA2_h_y_p_e_r_p_h]' |APC(S1386{ph} T1389{ph} S1390{ph} S1392{ph})
%obs: '[APC|rpt20AA3_h_y_p_e_r_p_h] | APC(S1501{ph} S1504{ph} S1505{ph} S1507{ph} S1510{ph}) |
\rightarrow S1510{ph}]

%obs: '[APC|rpt20AA4_h_y_p_e_r_p_h] | APC(S1562{ph} T1655{ph} S1656{ph} S1658{ph} T1661{ph} S1664{ph}) |
\rightarrow S1658{ph}]

%obs: '[APC|rpt20AA5_h_y_p_e_r_p_h] | APC(S1857{ph} S1861{ph} S1863{ph}) |
\rightarrow S1970{ph}]

\rightarrow S1970{ph}]

\rightarrow sub-a[3], sub-c[3], sub-r[1]

\rightarrow sub-a[3], sub-c[3], sub-r[1]

\rightarrow PP2_AA(sub-c[1] sub-r[3], PP2_B56A(sub-a[3] sub-c[2]) |

\rightarrow PP2_AA(sub-c[1] sub-r[3], PP2_B56A(sub-a[3] sub-c[2]) |

%obs: '[Cat-B_Total] | bCat()

%obs: '[Cat-B_S_3_3_un] | bCat(S33{un})

%obs: '[Cat-B_S_3_3_ph] | bCat(S33{ph})

%obs: '[Cat-B_S_3_7_un] | bCat(S37{un})

%obs: '[Cat-B_S_3_7_ph] | bCat(S37{ph})

%obs: '[Cat-B_T_4_1_un] | bCat(T41{un})

%obs: '[Cat-B_T_4_1_ph] | bCat(T41{ph})

%obs: '[Cat-B_S_4_5_un] | bCat(S45{un})

%obs: '[Cat-B_S_4_5_ph] | bCat(S45{ph})

%obs: '[Cat-B_L_x_un] | bCat(LysX{un}[.])

%obs: '[Cat-B_L_x_ub] | bCat(LysX{ub}[.])

%obs: '[Cat-B_naked] | bCat(S33{un} S37{un} T41{un} S45{un}) |
\rightarrow LysX{un}[.]

%obs: '[Cat-B_u_n_p_h_o_s] | bCat(S33{un} S37{un} T41{un} S45{un}) |
\rightarrow LysX{un}[.]

%obs: '[Cat-B_u_n_p_h_o_s] | bCat(S33{un} S37{un} T41{un} S45{un}) |
\rightarrow LysX{un}[.]
%obs: '[(Cat-B_u_n_p_h_o_s_u_b_i_q)]' | bCat(S33{un} S37{un} T41{un} S45{un}) LysX{ub}[
).
%obs: '[(Cat-B_p_r_i_m_e_d)]' | bCat(S33{ph} S37{ph} T41{un} S45{un})|
%obs: '[(Cat-B_h_y_p_e_r_p_h_o_s)]' | bCat(S33{ph} S37{ph} T41{ph} S45{ph})|
%obs: '[(Cat-B_d_e_g_r_o_n)]' | bCat(S33{ph} S37{ph})|
References


[47] Jiang, Z., Ninomiya-Tsuji, J., Qian, Y., Matsumoto, K., & Li, X. (2002). Interleukin-1 (IL-1) Receptor-Associated Kinase-Dependent IL-1-Induced Signaling Complexes Phosphorylate TAK1 and TAB2 at the Plasma Membrane and Activate TAK1 in the Cytosol. Molecular and Cellular Biology, 22(20), 7158–7167.


272
This thesis was typeset using XƎLATEΧ, based on Donald Knuth’s ΤΕΧ. The body text is set in 11 point Egenolff-Berner Garamond, a revival of Claude Garamont’s humanist typeface. The above illustration, Science Experiment 02, was created by Ben Schlitter and released under cc by-nc-nd 3.0. A template that can be used to format a PhD dissertation with this look & feel has been released under the permissive AGPL license, and can be found online at github.com/suchow/Dissertate or from its lead author, Jordan Suchow, at suchow@post.harvard.edu.