Modelling the evolution of biological complexity with a two-dimensional lattice self-assembly process

Alexander Stephen Leonard

Supervisor: Dr Sebastian E. Ahnert
Department of Physics, University of Cambridge

This thesis is submitted for the degree of
Doctor of Philosophy

Corpus Christi

March 2020

Declaration

This thesis is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the Preface and specified in the text. It is not substantially the same as any that I have submitted, or, is being concurrently submitted for a degree or diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. I further state that no substantial part of my thesis has already been submitted, or, is being concurrently submitted for any such degree, diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. It does not exceed the prescribed word limit for the relevant Degree Committee.

Alexander S. Leonard
March 2020

Acknowledgements

I would like to take this opportunity to express my immense gratitude for all those who have supported me on the journey to this moment. Firstly, I thank Dr. Sebastian Ahnert for his guidance and consistent advocacy. I never once felt limited in what questions I could ask or directions to explore, which led me to places I would never have imagined a few years ago. I am also grateful for the endless coffee and chats with my colleagues Will Grant, Marcel Weiß, Nora Martin, and Victor Jouffrey. Not only did they make the days more enjoyable, but many important ideas were born out of these idle discussions.

I would also like to thank Prof. Dimitri Vvedensky and Carolyn Salafia, M.D. for enabling and encouraging my first foray into the world of biologically-flavoured physics. At the time it was a departure into a strange new world, but now is a field I call home. Similarly, my academic path has been deeply inspired by Fred Crawford and Ian Coppell. My passion for science is due in large part to Fred, who frequently left me mesmerised with the beauty of simple physics demonstrations which comprehensively defied intuition. Although it took me many more years to appreciate, my penchant for coarse-grained reasoning was first implanted by Ian and his maps and models.

I am greatly indebted to Anna Platoni, Dan Grba, and David Allendorf, who not only provided friendship over the years, but also a place to stay when returning to Cambridge. I doubt these final months would have been possible without them. I am also grateful to the many friends in the Imperial College Underwater Club who have provided much needed respite over the past years. I am especially appreciative of the support from Katy Duncan, escaping the burdens of research and teaching over a Jack's Gelato come rain or shine.

I am eternally thankful of the loving support of my partner Dr. Olivia Ashton. She has been encouraging me since before applications, through the ups and downs of research, and I am glad she is with me at the end of all things PhD. I should also thank my two sisters, Chelsea and Sarah, for the love and support only siblings can provide. Finally, no set of acknowledgements would be complete without mentioning the unshakeable faith my parents hold for me. Their guidance and encouragement of my studies culminate in this moment. From entertaining my questions on the length of a string to refocusing my attention to maths classes, I have no greater inspiration in my life than my mum and dad. Thanks for getting me this far.

My methods are really methods of working and thinking;
this is why they have crept in everywhere anonymously.

EMMY NOETHER

If people do not believe that mathematics is simple,
it is only because they do not realize how complicated life is.

JOHN VON NEUMANN

One of the pleasures of looking at the world through mathematical eyes
is that you can see certain patterns that would otherwise be hidden.

STEVEN STROGATZ

Abstract

Modelling the evolution of biological complexity with a two-dimensional lattice self-assembly process

Alexander S. Leonard

Self-assembling systems are prevalent across numerous scales of nature, lying at the heart of diverse physical and biological phenomena. Individual protein subunits self-assembling into complexes is often a vital first step of biological processes. Errors during protein assembly, due to mutations or misfolds, can have devastating effects and are responsible for an assortment of protein diseases, known as proteopathies. With proteins exhibiting endless layers of complexity, building any all-encompassing model is unrealistic.

Coarse-grained models, despite not faithfully capturing every detail of the original system, have massive potential to assist understanding complex phenomenon. A principal actor in self-assembly is the binding interactions between subunits, and so geometric constraints, polarity, kinetic forces, etc. can often be marginalised. This work explores how self-assembly and its outcomes are inextricably tied to the involved interactions through the use of a two-dimensional lattice polyomino model.

First, this thesis addresses how the interaction characteristics of self-assembly building blocks determine what structures they form. Specifically, if the same structures are consistently produced and remain finite in size. Assembly graphs store subunit interaction information and are used in classifying these two properties, the determinism and boundedness respectively. Arbitrary sets of building blocks are classified without the costly overhead of repeated stochastic assembling, improving both the analysis speed and accuracy. Furthermore, assembly graphs naturally integrate combinatorial and graph techniques, enabling a wider range of future polyomino studies.

The second part narrows in on implications of nondeterministic assembly on interaction strength evolution. Generalising subunit binding sites with mutable binary strings introduces such interaction strengths into the polyomino model. Deterministic assemblies obey analytic expectations. Conversely, interactions in nondeterministic assemblies rapidly diverge from equilibrium to minimise assembly inconsistency. Optimal interaction strengths during assembly are also reflected in evolution. Transitions between certain polyominoes are strongly forbidden when interaction strengths are misaligned.

The third aspect focuses on genetic duplication, an evolutionary event observed in organisms across all taxa. Through polyomino evolutions, a duplication-heteromerisation pathway emerges as an efficient process. This pathway exploits the advantages of both self-interactions and pairwise-interactions, and accelerates evolution by avoiding complexity bottlenecks. Several simulation predictions are successfully validated against a large data set of protein complexes.

These results focus on coarse-grained models rather than quantified biological insight. Despite this, they reinforce existing observations of protein complexes, as well as posing several new mechanisms for the evolution of biological complexity.

Contents

List of Figures

List of Tables

CHAPTER

ONE

INTRODUCTION

Biology and physics were long separate pillars in the scientific world. Only at the tail of the 19th century did the two intermingle, with Karl Pearson seeding the concept of *biophysics*. Cooperation between these fields has driven advances that were inconceivable when viewed through a singular perspective, isolated in their traditional contexts. These developments necessarily rely on uniting concepts, approaches, and techniques across multiple disciplines, propelling areas as disparate as the mechanics of cancer [1] or foraging patterns of marine predators [2].

Interdisciplinary attitudes are essential to progressing science into a new era. The work presented in this thesis is situated at the interface of physics and biology, applying statistical modelling and analytic equations to the dynamics and evolution of complexity in protein quaternary structure.

1.1 Modelling biological complexity

Traditionally, much of science was based on *reductionism*, where understanding the minimal units of a system could constitute complete knowledge of the whole. This changed in the 1970's with the advent of complexity science. Complexity science revolves around *holism*, where the system can only be understood in its entirety; examining minimal units in isolation washes away the emergence of the dynamics of interest. Given the enormous complexity of biological systems, the language of complexity science finds a natural home in the context of biophysics.

Complexity science also emphasises the power of simplicity in modelling complex phenomena, known as effective theories [3]. As succinctly phrased by George Box, "all models are wrong, but some are useful" [4]. Distilling problems down to their most core concepts, at the expense of introducing approximations, enables the construction of new models. An imperfect understanding of a complex phenomenon is still a promising advancement in an otherwise intractable problem.

One such example is modelling atrial fibrillation, the largest cause of stroke, with a simple 3-state lattice model [5]. Elements in the lattice, i.e. heart muscle cells, can be resting, excited, or refractory, and can only interact with nearest neighbours. While this model is far too simple to capture *all* dynamics involved in atrial fibrillation, it captures *most* in a highly understandable framework. With orthodox clinical interventions enjoying little success, the approximate critical regions identified through the model for treatment represent a step forward.

Similarly, crowd dynamics leading to disasters have been modelled with barebone equations for the "force" exerted by adjacent pedestrians [6]. While the model may not provide directly implementable solutions on how to avoid such disasters, it is again a useful tool to explore ideas for potential solutions. This cyclic relationship between complex phenomena, coarse-grained modelling, and approximate understanding is highlighted in Figure 1.1.

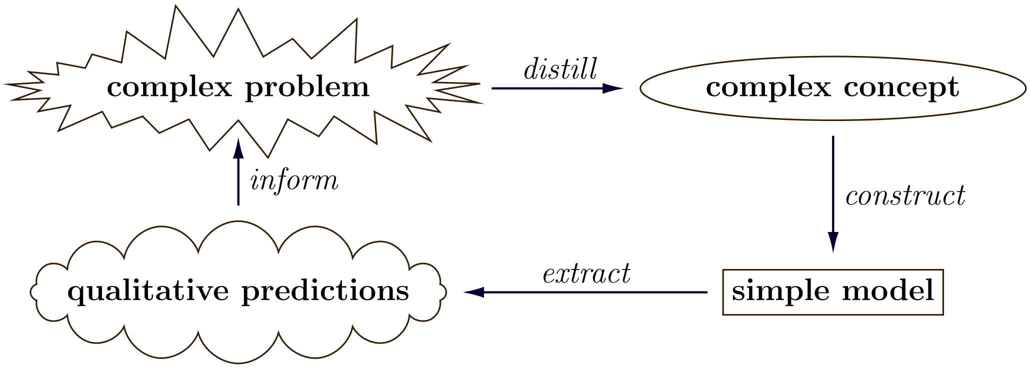


Figure 1.1: Complex problems often have details superfluous to the dynamics of interest, which can be expunged. A sufficiently focused concept can lead the construction of a coarse-grained model, which can generate qualitative predictions. These predictions do not provide quantitative answers, but can inspire novel interpretations or supplemental insights which may steer actionable solutions.

The Ising model [7] in physics is a prime example of the cyclic relationship. Determining the structural order of atomic dipole moments quickly grows unsolvable. By considerably simplifying the system into a rigid lattice and fewer states, the Ising model can calculate transition temperatures for ferromagnetism. Though all the predicted temperatures are inaccurate, the model correctly and generally predicts their existence (or not) for any dimension or geometry. If needed, more time-consuming models can then be used frugally to determine precise, quantitative results.

1.1.1 Coarse-grained proteins

Proteins are an elementary building block of life, participating in nearly every biological function. However, their ubiquity hinders a single "grand unified theory" of proteins, with multitudinous shapes, interactions, and functions. In the same vein as above, zooming out on the problem and taking a more coarse-grained view unravels universal underlying properties about proteins that would be lost in the noise of detail.

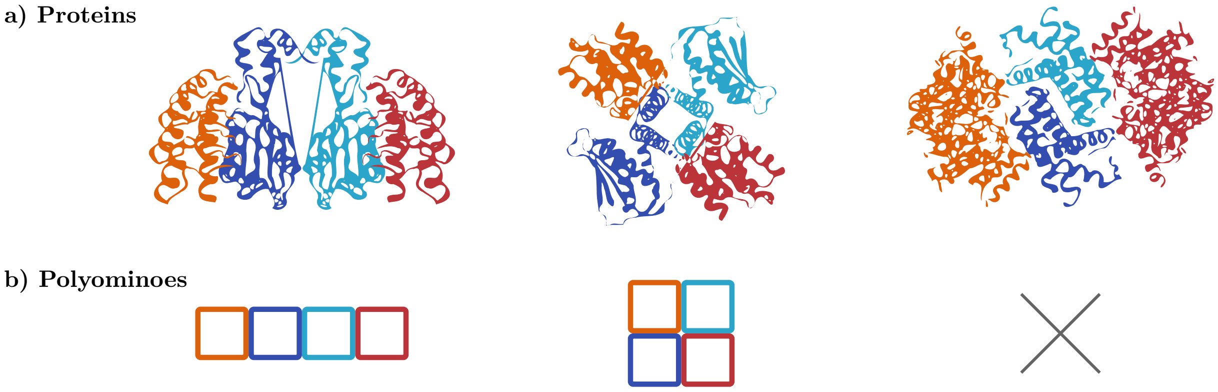
Protein complexes have four tiers of structure: primary, secondary, tertiary, and quaternary. Primary structure defines the long, linear sequences of amino acid residues. Secondary and tertiary structure are the conformations of local segments and the global three-dimensional fold of the protein subunit respectively. Multiple subunits can then arrange into a protein complex, which is the quaternary structure. Approximate models exist for various forms of lower level structure prediction, such as the Garnier-Osguthorpe-Robson (GOR) method [8] and hydrophobic-polar (HP) lattice folding [9], and their simple nature has been vindicated by their prediction success.

Primary, secondary, and tertiary structure are largely determined by energetic “folding funnels”. Quaternary structure still depends on Boltzmann distributions and thermodynamic stability, but principally emerges from the attractive binding interactions present between subunits. Different combinations of subunits can form different final structures, giving rise to the vast observed diversity of proteins. In addition to variability in subunit size, quaternary structures commonly vary from single subunits to several dozen, although some such as DNA viruses can contain thousands of repeated subunits.

Using a more coarse-grained perspective, many protein complexes become comparable. This observation has helped produce a “periodic table of protein complexes” [10], given evolutionary insight into the commonness of misassembly [11], and provided physical constraints on cellular function [12]. Polyominoes, geometric structures formed by adjoining uniformly-sized square tiles, are one such approach to this coarse-graining. Each tile within the polyomino corresponds to a protein subunit, and two adjacent tiles implies some attractive potential between the corresponding protein subunits.

Coarse-graining and polyomino representations can handle some of the geometric variation and size imbalance inherent in proteins. Interaction topologies can reasonably be detached from structural detail, and thus well-approximated, for many proteins. Some example proteins and possible polyomino representations are shown in Figure 1.2. However, polyomino models are not universally applicable. Protein quaternary structures can contain nuance that is lost when “blockified” to a lower resolution representation. Likewise, multi-site or cooperative bindings, where interactions can influence the binding affinity of others, do not translate well. Intrinsically disordered proteins or “fuzzy complexes” [13] are also outside the remit of lattice polyomino models. Despite these specific limitations, polyomino models can broadly approximate many classes of proteins and generate worthwhile results.

By carefully considering the approximations made and their impact, any intuition gained from the coarse-grained model can often be qualitatively applied to physical proteins. These coarse-grained models, in conjunction with experimental validation, continue to open new avenues of understanding protein dynamics.


a) Proteins

b) Polyominoes

Figure 1.2: a) Three examples of protein quaternary structures (PDBids: 6S9T, 6RFV, and 6OF2 from left to right), each composed of four protein subunits of two unique types. Subunits with similar colours indicate they are of the same unique subunit type (e.g. both blue-ish subunits have the same primary structure). b) Polyominoes are a coarse-grained model which approximate proteins, where each square tile in a polyomino imitates a protein subunit. Similar to above, tiles with similar colours are instances of the same unique subunit type. Some proteins, like the example on the right, have structures which cannot be reasonably approximated by uniform square tiles and are not suitable for a polyomino representation.

1.2 Mathematical models of tile self-assembly

Interest in structures formed by self-assembling sets of tiles dates back to at least 1961, starting with the question: can a set of squares with self-adhesive edges (Wang tiles) tessellate the plane [14]? Tile self-assembly has undergone substantial generalisations since then, increasing in scope and modelling potential. In 1973, Bennett established a link between Turing machines, effectively abstract “computers”, and biomolecules [15], paving the way for theories of biological computation.

Winfree provided substantial rigour 25 years later, introducing a biologically-flavoured algorithmic model of DNA self-assembly [16]. Recent work has focused on a complete computational infrastructure, formally showing how self-assembly spans various levels of abstraction. DNA-tile boolean circuits can form proof-reading structures, which in turn can form single-stranded tiles, enabling experimental algorithmic construction of DNA nanotubes [17]. Although these experiments are currently confined to laboratories, they conclusively prove that self-assembly is a multiscale phenomenon.

As the popularity of DNA-enabled computation grows, the number of distinct tile assembly models (TAMs) does as well. Each formal system is slightly tweaked to target a certain functionality, e.g. the kinetic TAM, reflexive TAM, two-handed TAM, signal-passing TAM, etc. Evans and Winfree reviewed the ever-growing plethora of TAMs, discussing how the TAM properties must align to experimental design considerations to have any meaningful physical realism [18].

Mathematical formalism of TAMs is a great strength, allowing results to be proven with complete rigour. With the gap bridged between theoretical self-assembling tile systems and experimentally possible DNA computers, this sound foundation of results is crucial. For example, a tile assembly system can be represented by $\mathcal{T} = (T, \sigma, \tau)$, where T is the set of assembling tiles, σ is the starting structure, and τ is the assembly "temperature". Terminal assembled structures are given by $A_t\left[\mathcal{T}\right]$. Grounding TAM results mathematically provides confidence while experimental validation remains encumbered by technological limitations.

Modelling self-assembly of coarse-grained proteins typically needs far fewer formal features to completely describe interesting dynamics. In particular, the potential for arbitrary computation through a set of self-assembling tiles is not currently relevant to dynamics of protein quaternary structure. As such, the polyomino model is a subclass of general TAMs, but specifically with relaxed requirements on cooperative binding. Regardless, results derived in a more relaxed model should not contradict the general theories established by previous TAMs—without careful consideration of the introduced approximations.

1.3 Lattice self-assembly in two-dimensions

The particular variation of tile self-assembly used as the foundation for this work is a minimalistic model. The polyomino model's core emphasis is on independent binding interactions that can take place between subunits. By eschewing physical complications of interaction geometry or thermodynamics, this lattice self-assembly model has enabled the study of diverse dynamics of protein quaternary structure.

In the decade or so since its inception, the model has found applications in algorithmic complexity and modularity of proteins [19], evolution of preferred structural symmetries [20], and genotype-phenotype maps of protein structures [21–23]. Biological applicability was secondary in most of these studies, as they focused primarily on exploring and growing the polyomino self-assembly model. Much of the following work results from these pioneering ideas, allowing greater effort to be spent on developing the conceptual linkage to existing protein quaternary structure literature.

While this work emphasises connections of the polyomino model to real protein dynamics, it should be underscored that polyominoes are an abstractified and idealised model of proteins. Some proteins are more amenable to this representation, like haemoglobins with approximately square geometry and equally-sized subunits, but polyominoes are not intended to serve as physical models. The purpose of this simplified model is to drill down on and isolate specific dynamics, and examine the contexts in which the observations can be translated back to proteins, seen primarily in Chapters 3 and 4.

1.3.1 Genotype representations and assembly graphs

At the most general level, a genotype is sequence of symbols drawn from a fixed-size alphabet which stores information about some higher level of organisation. For DNA, the alphabet consists of four nucleotide bases (A,T,G,C), while one possible form of a protein alphabet consists of twenty amino acids. In the specific context of this model, the alphabet describes the possible binding site types on the subunits, such that a genotype completely describes the set of self-assembling tiles. There are no limitations on what form the alphabet can take. Self-assembling systems only necessitate two ingredients: a characterisation of binding sites and an associated method for determining how two binding sites interact.

All existing uses of the polyomino lattice self-assembly model [19–23] use a set of integers as the alphabet. Interactions occur between sequential pairs of integers, e.g. binding site types **1** and **2** interact, **3** and **4**, etc. Certain binding sites are definitively noninteracting, e.g. **0** is always neutral.

Every group of four genotype elements (four faces on a square tile) forms one unique subunit type, encoding the binding sites with a clockwise convention starting from the top. Subunits can be represented with the notation $(-,-,-,-)$, where each $-$ is an integer interface type. Genotypes can encode one or more subunits with the notation $\{\ldots\}$. A full genotype encoding two subunits with a single interaction between them might then look like $\{(0,0,1,0),(0,2,0,3)\}$. Since the **4** interface type does not appear in this example genotype, the **3** interface type is effectively neutral.

The meaningful information contained within genotypes is the interactions between subunits, describing how different subunits can bind together. This information is best captured by an *assembly graph*, a compact description of all possible binding interactions between the faces of the subunits. Assembly graphs are the primary genotype representation used in this work.

In addition to offering a concise visualisation, assembly graphs are also rigorous. Specifically, assembly graphs can be interpreted as a compressed representation of edge-labelled pseudographs*. To preserve chirality, the internal structure of subunits is given by directed edges of one label, while interactions in the assembly graph are bidirectional edges of a different label. While the pseudograph representation is not necessary in self-assembly itself, the link with graph theory provides myriad insights covered in Chapter 2. An example of constructing an assembly graph and equivalent pseudograph from a genotype is shown in Figure 1.3.

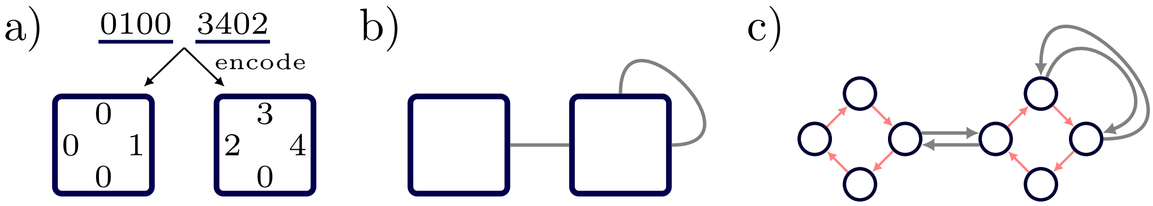


Figure 1.3: a) The genotype $\{(0,1,0,0),(3,4,0,2)\}$ encodes two subunits, starting clockwise from the top of each subunit. b) An assembly graph encapsulate all possible interactions, shown by grey lines, between and within the two subunits. c) This pseudograph is equivalent to the assembly graph in b), with bidirectional external edges. There are also internal edges (red) within each subunit, which maintain the geometric relation of the faces and preserves chirality.

*Pseudographs are explicitly multigraphs with loops, although some conventions already allow loops in a multigraph.

1.3.2 Polyomino assembly from an assembly graph

Genotypes and assembly graphs undergo the same assembly process for building structures. In the latter case, there is a more interpretable process. Rather than searching for a genotypic element on a given subunit to bind with, structure growth now essentially proceeds by walking on an assembly graph. The assembly algorithm is sketched out as:

- Seed the assembly with a random/fixed subunit.
- Assess all binding sites on the structure's perimeter for "open" interactions, i.e. there are possible binding partners to that site.
- Randomly select a subunit which matches an open interaction and bind it to the structure.
- Repeat previous 2 steps until no open interactions remain.

A more detailed algorithmic description is given in Appendix A.3. Subunits bind irreversibly to the growing structure, with no kinetic or temporal considerations allowing subunits to become unstuck[*]. During assembly, there is an infinite pool of each subunit type constructed from the genotype, avoiding any limiting reagent or rate-determining step dynamics.

The resulting lattice structure is a polyomino: a contiguous set of square tiles. Commonly, genotypes are assembled multiple times to comprehensively and accurately reflect possible behaviours. If any assembly does not terminate, i.e. always some open interactions remain, it is *unbound*. If the assemblies are not all identical, it is *nondeterministic*. Both of these properties can be undesirable in protein complexes or more general self-assembling structures, due to the inability to consistently produce the same assembly. An example of a deterministic and a nondeterministic assembly is given in Figure 1.4.

1.3.3 Phenotypes and polyomino equivalence

Judging if any two assemblies are equivalent is highly dependent on the context. Polyominoes have three levels of increasing strictness towards symmetry, where two polyominoes may be equivalent under one definition but not another. A *free* polyomino is the least strict definition, where a polyomino which can be reflected and/or rotated to match another is not distinct. *One-sided* polyominoes are only not distinct if there is some purely rotational operation, i.e. reflected copies are now distinct. A *fixed* polyomino is the most restrictive definition, where even a rotated copy of polyomino could be considered a new unique polyomino.

Regardless of the level of polyomino definition, comparing assembled structures revolves around grid-based bounding boxes. Each site in the grid details if it is occupied, along

[*] This is effectively a zero-temperature limit, and so this assembly approximates proteins that are thermally stable over long timescales.

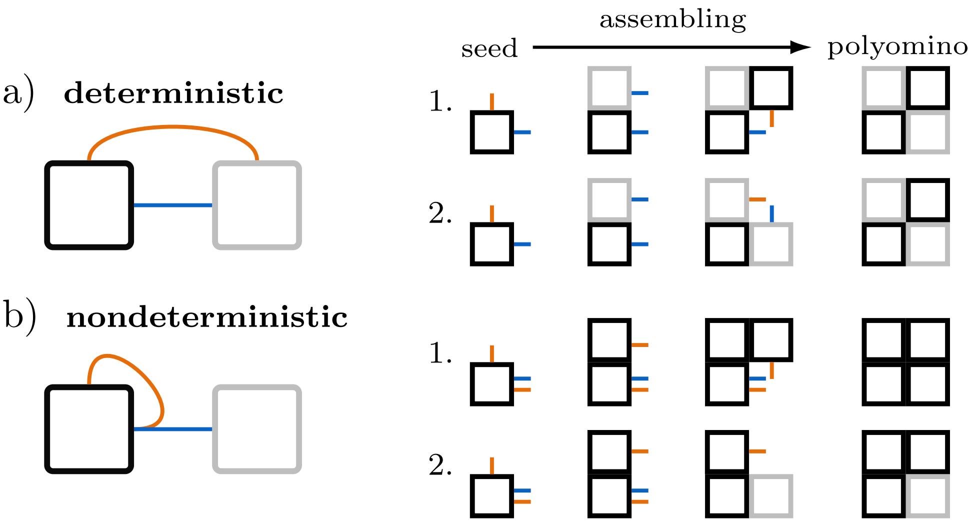


Figure 1.4: Some assembly graphs are deterministic, such as a), where multiple stochastic iterations build the same assembly (two iterations shown on the right). Others are nondeterministic, such as b), where the two assemblies have the same shape but differ by composition. Assembly graph subunits and polyomino tiles are colour-coordinated, such that e.g. a black tile placed during assembly indicates it is of the black subunit type from the assembly graph. Similarly, assembly graph edges and the open interactions in assembly are coloured.

with the subunit type and orientation. Preparing these grids so they can be compared for determinism within a single genotype or across different genotypes is challenging, due to inherent genotype symmetries. Extended details on this procedure can be found in Appendix B.

In addition to the different levels of polyomino definition, the contents of each grid site can be tuned to allow stricter comparison of two structures. Shape determinism is the most basic definition, where each lattice site is either occupied or empty. If the two lattice structures can be rotated/reflected as appropriate to match, they are shape deterministic. A stricter level is subunit determinism, where two structures must have the same shape as well the same subunit type in each lattice site. For an even finer granularity, the orientation of each placed tile can also be included when comparing structures as subunit + orientation determinism.

For example, two polyominoes may both form a two-by-two square, but one is formed of a single subunit type while the other is made of four distinct subunits. In the context of proteins, if only one subunit type could bind with an important ligand, these two complexes could have distinct functional roles. These two tetramers are shape deterministic, but would be meaningfully distinguishable under subunit determinism as two unique complexes.

The choice of polyomino definition and determinism level can appreciably alter any simulation or analysis and result in varying number of unique structures(see Figure 1.5). There is no universally correct choice; instead, the constraints should be appropriately chosen to best reflect the properties of the system under inquiry.

The phenotype of a genotype is defined here as the observable structure produced by the assembly algorithm. Bound and deterministic assemblies have an indisputable phenotype, clearly corresponding to the singularly produced structure. Nondeterministic

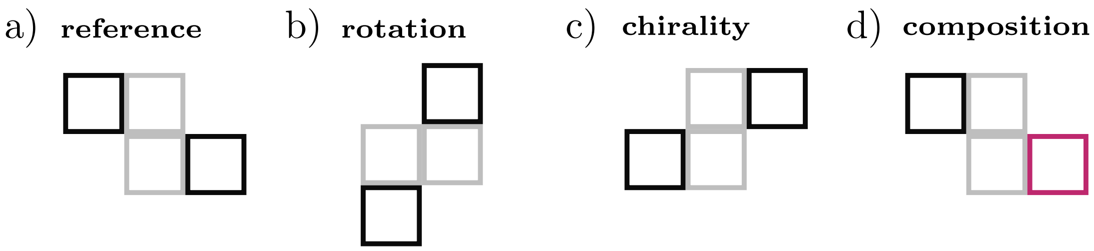


Figure 1.5: a) An example polyomino made of two distinct subunit types (black and grey). b) This polyomino is an identical copy to a) but rotated $\pi/2$ clockwise. These are two distinct *fixed* polyominoes, but are not distinguishable under *one-sided* or *free* definitions. c) Similarly, this reflected copy of a) is the same polyomino under the *free* definition, but is a different polyomino under the other two definitions d) Under subunit determinism, this assembly is distinct from a), b), and c) under any level of polyomino definition, as the red lattice site indicates a different subunit type.

assemblies can produce multiple polyominoes, ranging from single misassembled variants to never-repeating irregularity, injecting ambiguity into the definition of the phenotype. This thesis extensively uses the convention that the polyomino most commonly assembled from a genotype is assigned as its phenotype. A threshold can be implemented, such that a deleterious phenotype is assigned if no polyomino occurs above some frequency, for example 50%. Other methods might include always selecting the smallest or largest polyomino. Ongoing developments have extended the meaning of a phenotype to include the *set* of produced polyominoes, allowing a robust examination of nondeterminism, but this is still under investigation.

1.4 Thesis outline

The remainder of this thesis is structured as follows. Chapter 2 focuses on expanding an algorithmic approach to the classification of the determinism and boundedness for a set of subunits in the lattice self-assembly model. These two properties are critically important for any kind of self-assembly as they can lead to harmful protein aggregation or proteopathic misassembly. An algorithmic understanding of assembly behaviours also sheds light on the risks of repeated patterns in assembly and symmetry, which are more likely to give rise to unboundedness.

Chapter 3 introduces a generalisation to the lattice self-assembly model, with binary string binding sites that naturally establish an interaction strength. Selection pressures can drive the evolution of interactions to become stronger or weaker. These pressures arise when the order of assembly is important, and mirror experimental evidence that the order of evolution in complexes is reflected in the order of assembly.

Chapter 4 further expands the generalised model to include self-interactions, which allow greater modelling of symmetric homomeric and heteromeric interactions. Analytic results suggest symmetric homomeric interactions are quicker to evolve, while boundedness constraints imply heteromeric interactions are more evolvable. The duplication of homomeric subunits, which can then complementarily evolve into a heteromeric interaction between two subunits, is thus a highly efficient pathway for evolution. Experimental data taken from the Protein Data Bank supports several qualitative predictions of the model.

Finally, the conclusion summarises the salient results of this research, as well as covering some new areas of high potential for further study which may benefit from the developments of the generalised model.

CHAPTER

TWO

DETERMINISM AND BOUNDNESS OF ASSEMBLY GRAPHS

Key messages

- A new framework classifies assembly graph boundedness and determinism faster and more accurately than conventional stochastic assembly.
- Simple extensions to the assembly graph classification scheme expand the realm of dynamics that can be modelled.
- Assembly graph formalism solidifies existing empirical knowledge and facilitates further analytic results.

This chapter is based on the publication *Determinism and boundedness of self-assembling structures* by Tesoro, Ahnert, and Leonard [24]. Earlier iterations of this work were developed by Tesoro and Ahnert, and can be found in Chapter 4 of Tesoro's thesis [25]. Substantial improvements were made to the earlier work by Leonard and Ahnert, and unless otherwise stated, the material in this chapter is independent and novel work by Leonard under the supervision of Ahnert.

2.1 Introduction

Self-assembly is ubiquitous in nature, responsible for complex structures across multiple branches of science. Recent advances in synthetic biology, where custom sequences and structures can be made to order, have enabled designing self-assembling systems with biomolecules. DNA in particular has been proven to be extremely versatile in this regard, as specifiable DNA tiles can be used to construct complex nanoarchitectures [26]. Exploring the relationship between the "sticky edges" of these DNA tiles and the assortment of possible assembled shapes is still undergoing rapid growth [27, 28]. Similar attempts have been made to generate rule-sets for protein self-assembly [29].

Understanding how these simple units of assembly work together to form larger complexes is paramount to rational design. As established earlier, there are two focal properties of any general self-assembling system: boundedness and determinism. Boundedness refers to assemblies which are not unbound, i.e. terminate assembly with a finite size, while determinism means any genotype can only assembly a single structure. A biological complex which does not possess these two properties is at risk of potentially deleterious behaviour. An entire class of diseases, proteopathy, covers proteins with aberrant structures, e.g. aggregation, amyloid formation, misfolding, etc. Linking assembly dynamics to proteopathic origins may contribute to treating all manifestations of these diseases rather than case-by-case therapeutics [30, 31].

Conventionally, classifying these properties occurred through repeated stochastic assembly and checking for finite and identical structures. Developing a general framework for classifying the boundedness and determinism of arbitrary self-assembling structures is currently out of reach. In the following chapter, a specific framework for assessing the boundedness and determinism for assembly on a two-dimensional square lattice is introduced. Although the rules and analyses are directly applicable only to the polyomino model, several insights conceptually translate to observations about proteins.

2.2 Assembly graph classification scheme

Assessing the nature of assembly behaviour depends on features present within the prospective assembly graph: single interacting faces (SIFs), branching points, cycles, and symmetric subunits. Each of these features contributes known assembly dynamics, and so a classification scheme can be constructed which sequentially analyses an assembly graph for opportunities of unbound or nondeterministic growth. Any assembly graph which does not contain such features can therefore be classed as bound and deterministic.

The assembly graph classification scheme was pioneered by Tesoro and Ahnert [25]. Several improvements were required to streamline the assessment and, in some extreme cases, eliminate classification errors. Most mitigations focused on treelike SIF pruning, cycle detection, and nested cycles. These classification algorithm changes are highlighted in Figure 2.1.

The assembly graph analysis consists of sequentially applied rules which must be satisfied to classify an assembly graph as bound and deterministic. It does not fully account for the possibility of spatial conflicts, where two separate parts of an assembling structure might attempt to bind into the same lattice site. These occurrences are known as steric effects. Therefore, assembly graphs which are deterministic according to the classification algorithm rules must be subsequently checked for steric-determinism by assembling the structure a single time. The assembly graph is steric-nondeterministic if two distinct subunit types can occupy the same lattice site during this check, and is otherwise steric-deterministic.

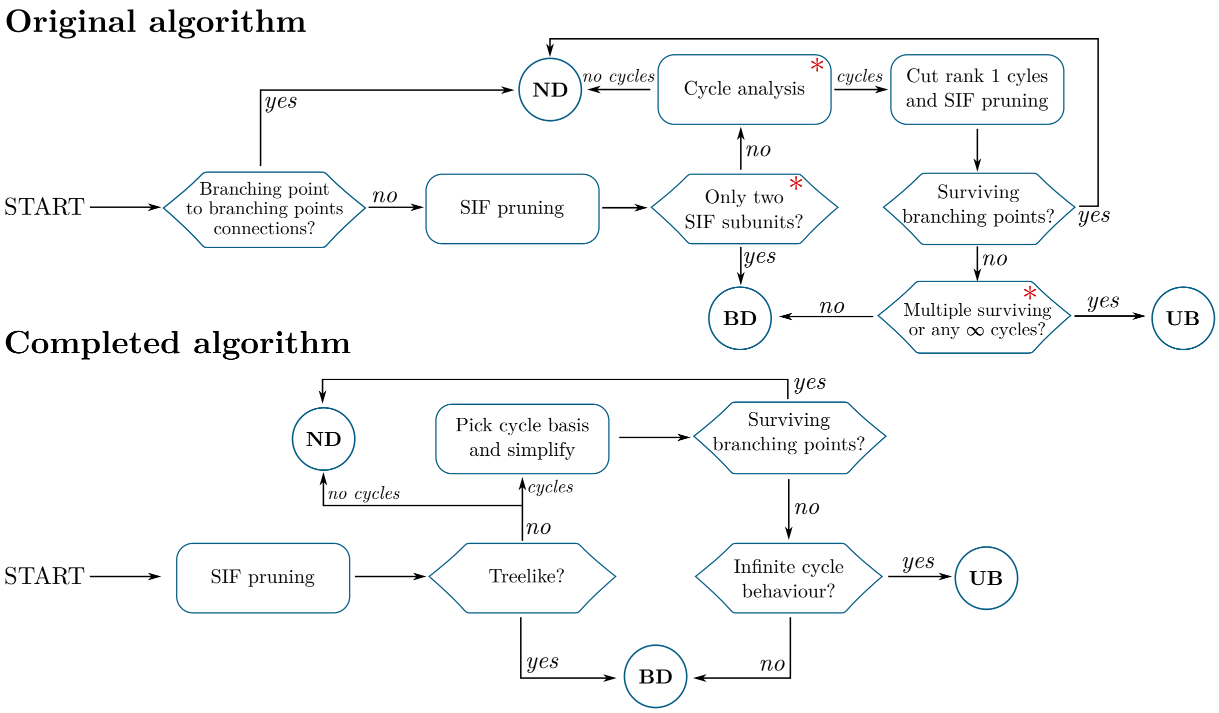

Original algorithm
START
Branching point to branching points connections?
yes
no
ND
no cycles
Cycle analysis
*
cycles
Cut rank 1 cyles and SIF pruning
SIF pruning
no
Only two SIF subunits?
*
Surviving branching points?
yes
yes
no
BD
no
Multiple surviving or any ∞ cycles?
*
yes
UB
Completed algorithm
yes
ND
Pick cycle basis and simplify
Surviving branching points?
no cycles
cycles
no
no
START
SIF pruning
Treelike?
Infinite cycle behaviour?
yes
UB
yes
BD
no

Figure 2.1: The original assembly graph classification scheme [25] had several problematic steps which led to misclassifications, indicated by red asterisks. The final classifications possible are bound and deterministic (BD), nondeterministic (ND), and unbound (UB). Much of the scheme was reformulated to eliminate these problems and simplify the logic involved in each classification step. Input for both algorithms is the same, taking in assembly graphs already tested for connectedness.

Classifying assembly graphs in this manner has theoretical importance, contributing general insight into sources of irregular assembly. However, the associated gain in classification speed also has powerful implications on what new systems can be examined, expanding analysis to larger genotype spaces. As such, refining the logic present in the initial scheme plays a crucial role in maximising performance in addition to correcting errors.

2.2.1 SIFs and deterministic treelike graphs

Single interacting faces are a common assembly graph feature, where a subunit only has one face with interactions and three noninteracting faces. Subunits with this feature tend to be “ornamental”, binding to the periphery of a growing structure. Once attached to a growing structure, a SIF subunit has no remaining interactive sides and so terminates assembly in that particular direction. Unbound growth requires a constant addition of new interactive faces on the structure’s perimeter, and so SIF subunits **cannot** induce unboundness. Similarly, SIF subunits cannot be nondeterministic as they do not introduce any further assembly choices, let alone multiple. Due to their relative simplicity, SIF subunits are the most straightforward feature to analyse.

A *treelike* assembly graph is defined as an assembly graph which does not contain any cycles or non-SIF branching points. Branching points occur when multiple binding partners exist for a given face, e.g. in the genotype $\{(0,1,0,0),(2,2,0,2)\}$ where the **1** can bind with either of the two **2** faces. Such assembly graphs can be re-expressed with SIF subunits and single-use interactions, as seen in Figure 2.2. A subunit without any binding interactions is the simplest example of a treelike assembly graph, similar to the simplest tree graph which is a single node and no edges. Such an assembly graph is evidently bound and deterministic.

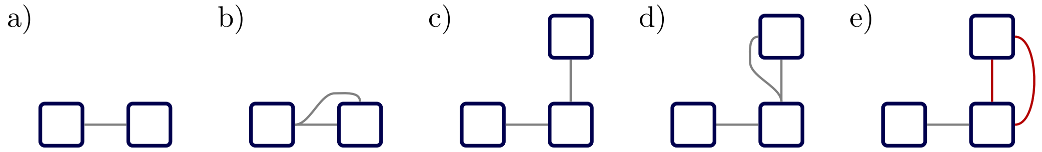


Figure 2.2: a) The simplest treelike assembly graph with an interaction is two connected SIF subunits. b) Assembly graphs with SIF branching points are can still be treelike. c) The assembly graph from b) can be re-expressed without the branching point using additional SIF subunits. d) Non-SIF branching points prevent assembly graphs from being treelike, but they do not necessarily cause nondeterminism. e) Assembly graphs with cycles, highlighted by red edges, also cannot be treelike.

Adding only a SIF subunit, potentially a SIF branching point, with a new interaction to the assembly graph cannot impart any non-SIF branching points or cycles. Therefore adding a SIF subunit to a treelike assembly graph leaves it treelike. Since the smallest treelike assembly graph is bound and deterministic, and adding SIF subunits cannot contribute behaviour to the contrary, then by induction any treelike graph must be bound and deterministic.

During this iterative construction, adding SIF subunits can transform earlier SIF subunits into subunits with multiple interacting faces. While the induction argument still holds, in practice this makes identifying an entire assembly graph as treelike difficult. Removing SIF subunits and their interactions wherever possible, i.e. “pruning” the assembly graph, can restore these multiple face subunits to being SIF. Eventually, any subunit that was iteratively constructed as a SIF will have been removed.

After sufficient SIF subunits have been recursively pruned, the assembly graph should be more obviously treelike or unable to reach a treelike form, thus being bound and deterministic or needing further analysis respectively. This pruning procedure is shown in Figure 2.3.

The pruning process is preserved from the initial classification scheme [25], but the treelike assembly graph definition was expanded. Now a single pruning condition can be used on all assembly graphs with fewer contingencies. Although a trivial case, a subunit without any interactions can now also be correctly identified as treelike and thus bound and deterministic.

2.2.2 Fundamental cycle basis

Cycles are *walks* on an assembly graph, alternatively stepping between external edges (subunit interactions) and internal edges (subunit faces), returning to the initial location

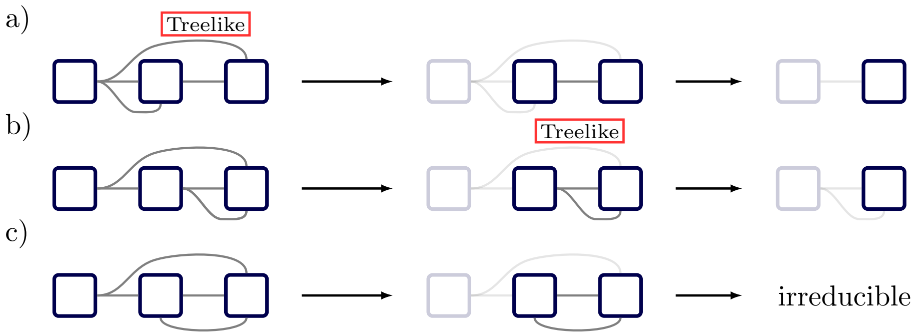


Figure 2.3: Both the assembly graphs in a) and b) are ultimately able to be pruned into a subunit without any interactions, and so are bound and deterministic. However, the assembly graph in a) is identifiable as treelike (no cycles or non-SIF branching points) immediately and could be classified without further pruning. On the other hand, the assembly graph in b) requires one round of pruning to reach an obvious treelike state. Although the SIF branching point in c) can be pruned, the assembly graph cannot be reduced to a treelike state as it contains a cycle, and so must continue into the next round of classification.

without repeating any edges. A key property of an assembly graph cycle is its rank, or how many times the involved subunits are reused before the cycle completes its assembly [24]. Cycles can take ranks of 1, 2, 4, or ∞, with some examples shown in Figure 2.4 highlighting the correspondence between cycle rank and subunit repetition. Rank ∞ cycles correspond to directly unbound growth, as that pattern can recur infinitely without terminating.

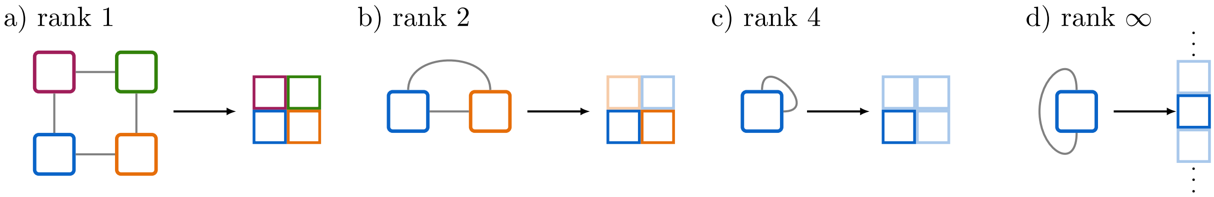


Figure 2.4: a), b), and c) all produce two-by-two tetramers, but with varying repetition of subunit types, which corresponds to the cycle's rank. Subunits and tiles are coloured as in Figure 1.4, and re-used subunits are indicated by faded tiles to highlight the nature of the rank. d) This cycle continues forever using the same subunits (in this case just one subunit), indicated by the dots, and hence is rank ∞.

Identifying assembly graph cycles is trivial when each edge in a cycle is exclusive to that cycle. When cycles share edges with other cycles (or branching points), the situation becomes more complicated. The symmetric difference (union minus intersection) of two cycles is itself another valid cycle. This induced cycle does not contribute any assembly behaviour not already contained within the two original cycles; rather it is a mixture of them. Just as a vector space has a basis formed by a minimal spanning set, an assembly graph has a cycle basis. Every possible cycle present in an assembly graph is then some combination of cycles within its basis.

Since only cycles in a basis contribute unique assembly behaviour, only the cycle basis should be analysed for detecting unbound potential. Unfortunately, constructing a minimum cycle basis, such that each cycle is as short as possible, is an NP-hard problem in graph theory. Additionally, not all graphs have a unique minimum cycle basis [32]. Any basis is sufficient for detecting unbound growth, but a minimal basis is the most efficient.

There is no general algebra for combining the ranks of the combined cycles. For example, two rank 2 cycles can join together to form a rank 1 or rank ∞ cycle, depending on the exact configuration. Only the latter case leads to unbound growth, and so these two cases contribute completely different behaviour. However, combining a rank R and a rank 1 cycle always produces another rank R cycle.

The main advancement from these two observations is that rank 1 cycles can **always** be cut and pruned without affecting the fundamental behaviour of an assembly graph. Regardless of whichever edge is cut, the surviving cycles will still form an equivalent cycle basis. The ranks of the cycles in the basis may change depending on the cut, but their combinations will preserve the potential of unbound growth if it previously existed. Some example cycle bases are shown in Figure 2.5, including a rank 1 cycle case.

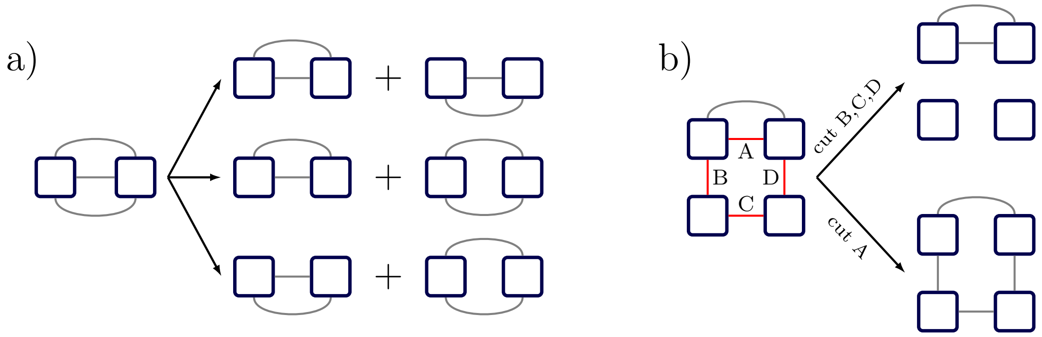


Figure 2.5: a) This assembly graph contains two rank 2 and one rank ∞ cycle, all involving two edges. There are three choices for the cycle basis, and no one choice is more fundamental. However, the latter 2 choices containing a rank ∞ cycle are easier to directly identify as unbound. b) Assembly graphs with rank 1 cycles, highlighted with red edges, must be cut and pruned before selecting the cycle basis. Cutting edges B, C, or D lead to a different basis than cutting edge A. However, every choice has equivalent behaviour (but not necessarily equivalent efficiency).

2.2.3 Spatially nested cycles

Unbound growth can only occur from continuously reusing some subset of assembly subunits. This behaviour commonly derives from one cycle completing its assembly, typically leaving exposed interacting faces belonging to another cycle, which after assembly leaves new faces belonging to the original cycle exposed and so on. Identifying multiple irreducible cycles is thus tantamount to identifying unbound growth and classified as such. However, spatial effects can invalidate the seemingly trivial previous axiom that multiple cycles assert unbound growth [25].

Multiple cycles can be spatially "nested", such that the completion of one cycle coincides with the completion of another cycle, leaving no new exposed interactions. When nesting occurs, and cycles do not regenerate each other, the assembly remains bounded. An example of cycle nesting is shown in Figure 2.6, demonstrating that minute differences at the assembly graph level can have huge impacts on the assembly behaviour.

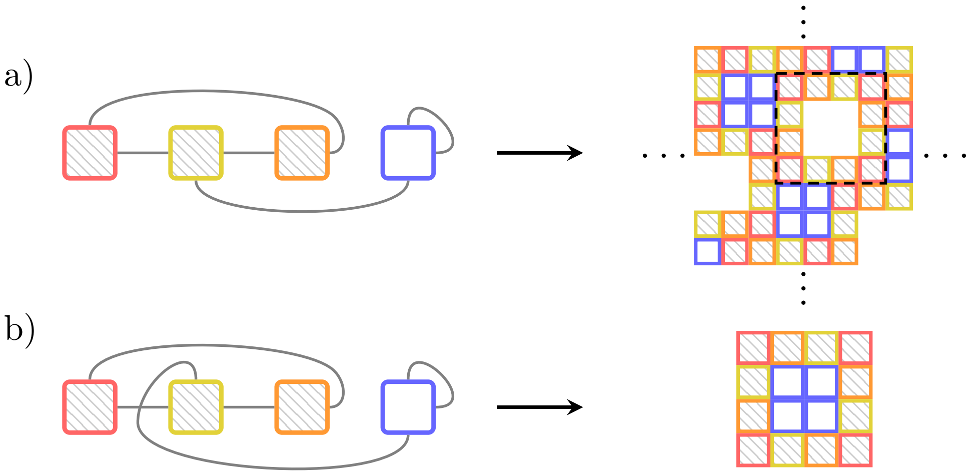


Figure 2.6: Assembly graphs in a) and b) only differ by where the blue subunit interaction connects to the yellow subunit. Assembly is unbound in the former but bound in the latter. This results from the hatched red-orange-yellow (ROY) cycle in the former case exposing new open interactions for the blue cycle, eventually regenerating exposed faces for the ROY cycle and so on. In the latter case, all initially exposed faces are bound to by the end of a single assembly cycle, leaving no possibility for further growth. For clarity, a complete assembly iteration of the ROY cycle is demarcated by dashed lines in a), which is equivalent to the same ROY cycle in b). See Figure 1.4 for details on colour notation.

As a spatial effect, cycle nesting cannot be determined solely through assembly graph analysis. Even cycles of different ranks can be embedded within one another, making nesting difficult to identify by empirical rules. Instead, returning to the fundamental definitions of cycles and unbound growth provides the important test. Since, by definition, a rank R cycle repeats the assembly pattern R times, the maximum expected bound size B of any polyomino produced from the assembly graph is then:

$$B = \sum_{c \in \text{cycles}} R_c \cdot L_c + E$$

where R_c is the rank of cycle c and L_c is the number of subunits involved in that cycle. All SIF subunits should be pruned prior to this analysis, but the contribution of any non-prunable subunits external to cycles is included as E. Should the attempted lattice assembly exceed this size, or identify a subunit in a cycle being used more than R_c times, the cycles are not nested and thus growth will be unbound. In principle, B can be minimised by optimally cutting rank 1 cycles and preserving the minimal cycle basis, i.e. smallest L_c. In practice however, any time advantage gained through using the optimal cut is outweighed by the time required to determine the optimal cut, and so greedy algorithms are more effective.

2.2.4 Cycle detection heuristics

Constructing a cycle basis for an assembly graph is one of the most intensive steps of the classification scheme. Although there are many interesting exceptions, assembly graphs with many edges relative to the number of subunits are typically unbound or nondeterministic. Most previous polyomino studies [19–21] have focused explicitly on bound and deterministic assemblies, implicitly lumping unbound or nondeterministic assemblies into a single generic deleterious phenotype. Merging these categories allows several probabilistic alternatives to constructing the cycle basis.

Topological graph theory can provide inspiration for one such heuristic. The first Betti number b_1 identifies the number of topological "holes" in a graph. This translates without modification to assembly graphs, and is calculated as $b_1 = m - n + 1$ for an assembly graph with m interactions and n subunits. Since branching points are typically either removed earlier or lead directly to non-determinism, b_1 is a useful predictor for the number of cycles in pruned assembly graphs. Values of zero or one typically correspond to valid phenotypes, while a value of two is borderline. For values greater than two, assembly graphs overwhelmingly will correspond to the deleterious phenotype.

Similarly, the Cheeger constant measures the lack of "bottleneckedness" in a graph, i.e. the resilience that a single cut will not bisect the graph. Estimating a high Cheeger constant for an assembly graph implies high linkedness and thus likely nondeterminism or unboundedness, while a low value implies treelikeness and thus determinism.

Converting these tools from graph theory has the additional bonus of being well-studied and easily available through open-source implementations. Specific studies may have a contextual basis where analysis speed gain supersedes precision, allowing integration of one or more heuristic algorithms. For example, with two subunits and eight interface types in the polyomino model, only 2% of valid phenotypes are misidentified using $b_1 \geq 2$ as a rejection criterion but the analysis is nearly a third faster. However, larger genotype spaces are capable of more intricacy which can disrupt these heuristics, although this has not been tested in depth.

2.3 Classification scheme extensions

Correctly identifying the boundedness and determinism of self-assembling structures is not simple, and the classification scheme is correspondingly rigid to ensure accuracy. However, there are several extensions which augment the range of application without incurring much, if any, overhead. These extensions overcome the inflexibility of the classification scheme with regard to many biologically relevant features of self-assembling systems, such as self-interactions and non-constant interaction pairings.

2.3.1 General interactions

Self-interactions have rarely been considered in polyomino models, only receiving secondary interest in one previous study [21]. However, the inclusion of self-interactions does not require any conceptual modification to the classification scheme, and increases the diversity of phenotypes achievable from a fixed number of subunits. Self-interactions, by definition, bind to themselves. As such, these interactions are an additional type of intra-tile cycle of rank 2. Otherwise, the classification scheme itself is entirely unchanged and proceeds as usual.

Note that the presence of more than one self-interaction necessarily implies branching points connected to branching points, overlapping cycles and branching points, or infinite cycle growth. While it is possible that such assembly graphs build a valid phenotype, self-interactions generally should appear sparingly. This is not always the case in real proteins, where multiple self-interactions frequently occur due to more detailed interaction modes and geometric constraints in higher dimensions, such as chiral binding etc.

2.3.2 Seed dependence

Examining assembly behaviour as a whole rather than depending on seed choice is a strength of the assembly graph scheme. There are contexts where assembly may always start from a particular seed tile (or a set of such tiles). It is possible for an assembly to be deterministic from certain seeds but not from others. If, for example, one genetic sequence was always transcribed and translated into a protein subunit before other involved subunits, it may be deterministic despite the overall assembly graph being nondeterministic or unbound. In the same example, it is also potentially meaningful to understand that the assembly is vulnerable to the translation order, perhaps giving insight into certain forms of protein misassembly.

Nondeterminism primarily creeps into assemblies through branching points which cannot be removed during the SIF pruning procedure, which represent faces that can bind to more than one other face. However, the assembly process only has this choice if the seed and initial structure are "upstream" of the branching point. If the assembly starts "downstream", the assembly only encounters the branching point after the choice of binding tile was determined by an earlier assembly step. An example of this behaviour is shown in Figure 2.7.

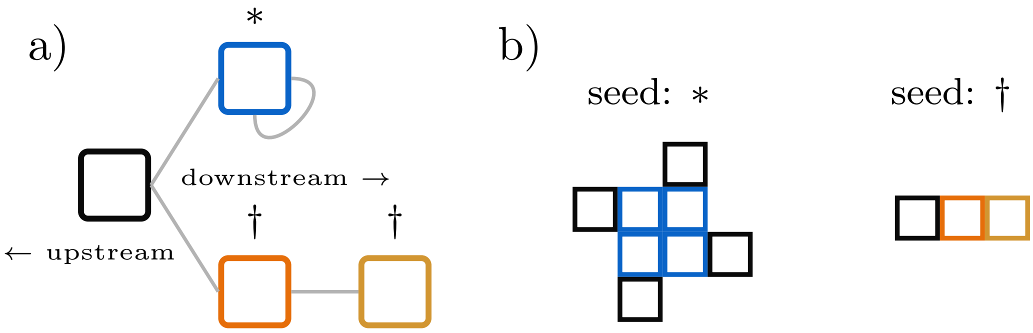


Figure 2.7: a) Pruning the SIF branching point in this assembly graph would leave the assembly graph disconnected and so is nondeterministic. However, this disconnected nondeterminism does not occur from all seeds, and specifically only the black subunit. b) Assembling with the $*$ or † marked subunits as seeds leads to deterministic assembly, forming a 'Catherine wheel' or 'I'-shaped trimer respectively. Both of these cases are seeded downstream of the branching point, and so never encounter nondeterministic branching possibilities during assembly. Colour notation is the same as Figure 1.4.

This quasi-deterministic behaviour can be generally detected through an additional step to the SIF pruning procedure. Disconnecting SIF branching points are a sign of general nondeterminism, but can be exploited to add seed dependence. After pruning the assembly graph, any disconnected components not containing the chosen seed(s) should be discarded. The classification on the remaining partial assembly graph will correspond to the classification of the overall assembly given those fixed seeds.

More complex cases may require a directed trail on the assembly graph starting from the seed of choice. If no branching points were encountered by the end of the trail, then that seed is deterministic. Checking for these walks then replaces the "surviving branching points" assessment step in Figure 2.1.

2.3.3 Variable interaction environments

Another aspect disregarded by the standard classification scheme is phenotype plasticity, an important contributor to evolutionary adaptations. Plasticity is effectively a phenotype-environment coupling, where changes in the environmental context are reflected by mapping to different phenotypes. This can trivially be encapsulated by partitioning or creating separate interaction rules, e.g. a high pH context where **1** and **2** interact and a low pH context where **1** and **3** interact. Assessing interactions which are not always active in all contexts will lead to overly-strict nondeterministic or unbound classification, so considering environmentally relevant subsets is useful.

Alternatively, interaction rules can be applied sequentially, e.g. cooling an assembly stabilised at some high temperature before mixing with subunits which can only bind at lower temperatures. If the assembly graphs formed under each set of interaction rules are bound and deterministic, and the steric assembly test follows the sequence of interaction rules without issue, then the overall assembly process is also bound and deterministic. Assembly graphs can support more interactions before risking unbound or nondeterministic growth under these regimes.

While these situations may rarely occur in standard protein formation, this concept of sequential interaction rules could prove useful in contexts such as nanofabrication. Structures can be assembled over time that would otherwise be nondeterministic or unbound if all interactions or subunits were active simultaneously. Furthermore, structures could be designed to assemble the inner core first before appending outer layers. This process of ordered assembly is returned to in Chapter 3.

2.4 Advantages of assembly graphs

Stochastic assembly approaches, as used in all previous polyomino studies, are generally acceptable for specific applications. Classification accuracies and speeds are reasonably high provided the number of repeated builds K is appropriately chosen for a context. Exact implementation of the self-assembly algorithm rarely impacts classification accuracy as well. However, all of these (potentially minor) detractions are entirely allayed by the assembly graph classification scheme.

2.4.1 Classification accuracy

One of the major problems of classifying properties based on repeated stochastic assembly is probabilistically missing an assembly which would change the classification. Such an event is reminiscent of a "black swan" [33] in other fields, where an extremely rare situation can have extreme consequences. Consider a biomedical engineering context, where a hypothetical structure should be strongly bound and deterministic. Even though the probability of misclassifying a self-assembling system may be minimal, the consequences of unbound aggregation in the heart or brain could be severe.

It is relatively straightforward to construct obvious examples where nondeterministic or unbound growth might be missed. There are more convoluted cases where the misclassification rate is higher, but the principle is the same. The easiest case is a three subunit system effectively forming a "long" chain with the infinite cycle subunit $(1, 0, 2, 0)$ that is stochastically terminated by the "caps" $(1, 0, 0, 0)$ and $(2, 0, 0, 0)$. Any finite or semi-infinite chain can be assembled from any seed, except for chains of length two, which only form if a cap is used as the seed, as seen in Figure 2.8.

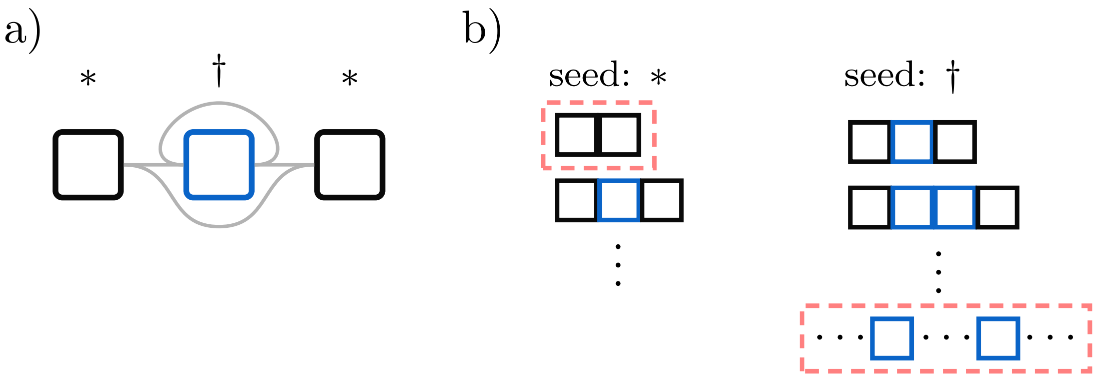


Figure 2.8: a) The assembly graph for the three subunit system mentioned in the main text, where the two black subunits have identical assembly behaviour. b) Only two structures cannot be assembled from any seed, shown in red boxes: the dimer (only seeds ∗), and the infinite chain (only seed †). Colour notation follows Figure 1.4.

Regardless of initial seed, the probability of forming an N-tile chain with this assembly graph is $P(N) = (1/2)^{N-1}$ for $N \in [3, \infty)$. Generally, probabilities differ per seed, but in this carefully chosen example the probability is independent of seed over the above range of N. Hence the probability of getting the same N-tile chain after K repeated builds, that is to say misclassifying this nondeterministic and potentially unbound structure as bound and deterministic, is then:

$$e(K) = \sum_{N=3}^{\infty} \left[(1/2)^{N-1}\right]^{K}$$

Misclassification of this assembly is negligible, $e\left(10\right) = 10^{-6}$, and becomes exponentially rare with growing K. However, the chance of failure is inherent in the stochastic approach, and can only be offset by tediously comparing more and more assemblies. More complicated assembly graphs, or weaker determinism criteria, can hugely inflate the error rate. This can only be compensated by increasing K, which is a direct tradeoff with analysis speed.

Seed dependency and lower bound of K

Nondeterminism and unboundedness typically arise from the combined behaviour of a subset of subunits. Rarely do these behaviours equally involve every subunit within a genotype. As such, seeding assembly with certain subunits can lead to significantly different outcomes, as seen earlier in Section 2.3.2. For general genotypes, a structure must be built at least as many times are there are subunits, $K \geq N_S$, in order to exhaustively test for undesirable assembly outcomes. While most genotype lengths of interest are relatively short, with $N_S = 4$ one of the largest studied so far, such a constraint inherently limits the exploration of larger genotypes.

In addition, as highlighted in the extensions section, overall nondeterministic or unbound assembly graphs can display bound and deterministic behaviour from certain seeds. The reverse implication here is that stochastic assembly can classify overall deleterious assemblies as valid from certain seeds. More pressingly, the meaningful number of repeated builds can be significantly less than K. Seeding from any bound and deterministic seed contributes zero additional classification power, wasting those assemblies. As the effective K drops for larger assembly graphs, the error rate again can inflate rapidly.

For these genotypes, a reasonable condition on the number of builds is $K/N_s \gg 1$, to ensure that each seed is amply tested for misassembly. Such a condition again directly introduces a conflict between accuracy and speed, which is not present in the single run of the assembly graph framework. Without any *a priori* knowledge of which subunits may contribute the most classification power, the inelegant solution is just assemble a lot.

2.4.2 Unbound growth identification

Unbound growth can be a highly destructive phenomenon, for example when misfolded proteins aggregate in an uncontrolled way. Empirically derived bounds have been used in previous studies [20, 21], where a structure is assigned an assumed unbound classification if growth extends sufficiently in any one direction. Since most studies were limited to short genotypes, the upper bounds only had to be found for several genotype lengths.

With the assembly graph formalism, it is trivial to establish an upper bound on bound polyomino size. In the case of a single subunit, a rank 4 cycle produces the largest polyomino, with size 4. For larger genotypes, the additional subunits can form (sterically self-limiting) rank 4 cycles on the growing perimeter, giving a total size of $4N_S^2$. These assemblies are typically only seed-deterministic, so have appeared in these previous studies but are flagged as nondeterministic by the assembly graph scheme.

A better approximation for bound and deterministic structures uses rank 4 cycles consisting of multiple tiles. Using all the available subunits in a single cycle leaves a large hollow centre, so using some subunits as SIFs is beneficial. Using L_c of the N_S subunits in a rank 4 cycle gives a deterministic polyomino of size:

$$S = 4\left(N_S\left(L_c + 1\right) - L_c^2\right)$$

The optimal split is found through the derivative:

$$\frac{dS}{dL_c} = 4N_S - 8L_c = 0 \rightarrow L_c^* = \frac{N_S}{2}$$

which gives a polyomino of size:

$$S^* = N_S^2 + 4N_S - \sigma \text{ where } \sigma = \begin{cases} 0 & \text{if } N_S \text{ even} \\ 1 & \text{if } N_S \text{ odd} \end{cases}$$

where σ corrects for L^* requiring an integer form. Even larger deterministic shapes are possible, but come highly convoluted assembly graphs of nested branching points which have no general pattern. However, these two expressions form relatively tight bounds on the largest deterministic polyomino possible given N_S subunits, as shown in Figure 2.9.

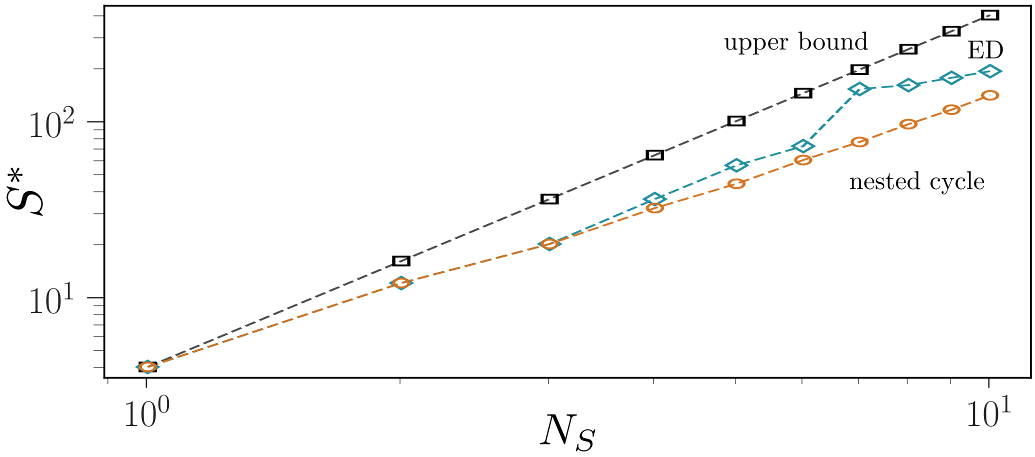


Figure 2.9: The maximum polyomino size S^* grows rapidly with increasing subunits. Genotype spaces quickly grow too large to search exhaustively, and so evolutionary dynamics (ED) were used to empirically determine maximum size. The analytic form of the upper and lower bounds are good approximations, and are derived from first principles.

Importantly, as nondeterminism and unbound growth can commonly co-occur, the assembly graph classification scheme can be used to identify unbound growth through the cycle analysis. In the stochastic assembly approach, few unbound and nondeterministic assemblies will hit the threshold for unbound growth through probabilistic self-limiting growth. As such, they are likely classified as nondeterministic but bound. If unbound growth and nondeterminism are merged into a single deleterious phenotype, this misclassification is not problematic. Regardless, genotypes can now be correctly categorised with finer granularity under the assembly graph scheme, allowing more detailed analyses.

2.4.3 Classification speed

Self-assembly within the polyomino model is not computationally intensive. A single assembly may occupy on the order of tens of lattice sites, tracking a similar quantity of possible bindings. Storing the results for K repeated assemblies likewise requires inconsequential amounts of computational memory. Stochastic classification of a single genotype happens near-instantly. However, the genotype spaces are also vast, ranging from 10^6 to 10^{12} in previous polyomino studies [21, 23]; accelerating the classification speed can be critical to exploring larger spaces.

Stochastic assembly and assembly graph classification are two fundamentally different approaches. The former relies on detecting differences in repeated lattice assembly, while the later focuses on graph features that are known to generate irregular behaviour. To compare their relative speeds, the same set of sample genotypes were classified for various genotype lengths N_S. Genotypes which did not form a single connected component were rejected. Available interaction types scaled with genotype length as $4N_S + 2$, ensuring all possible bound and deterministic assembly graphs could form without overly inflating the genotype space. Classification speeds for both approaches are shown in Figure 2.10, with both having been efficiently re-implemented in modern C++ to minimise confounding factors.

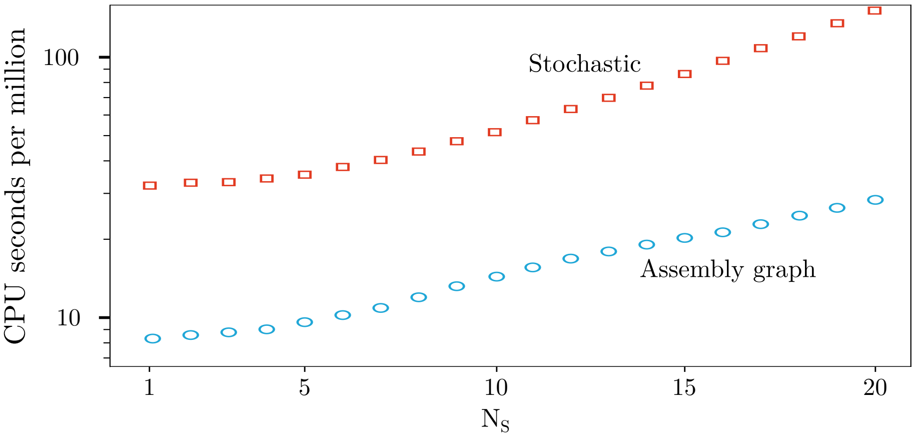


Figure 2.10: Algorithm efficiency depends on the context, e.g. genotype-phenotype maps have far fewer bound and deterministic assemblies to classify than evolutionary dynamics, but the assembly graph approach outperforms stochastic classification (with $K = 10$) over random genotypes by an average factor of ~ 3.5. As genotypes lengthen, structures typically enlargen, asymmetrically affecting stochastic classification while assembly graphs scale better.

Although assembly graph classification handles longer genotypes better than stochastic classification, the effect can be misleadingly understated. More and more of the genotype space maps to unbound or nondeterministic phenotypes as genotype length increases. Since stochastic classification can “short-circuit”, i.e. classify immediately when nondeterminism or unboundedness is encountered, these longer genotypes rarely last until the last build. So even with some fraction of assemblies requiring less than 10 repeated builds, assembly graphs still outperform stochastic classification.

Evolutionary dynamics, which primarily focus exclusively on bound and deterministic phenotypes, showcase this issue clearly. Here, most genotypes will require the full K assembly attempts, entailing maximum compute time. Compared with the general factor of ~ 3.5, assembly graphs gain a speedup factor of approximately 10-15 over stochastic classification in the specific context of evolution simulations. Similarly, speedup gain continues to grow as genotypes lengthen.

More problematically for stochastic classification in evolutionary dynamics, the number of builds K is fixed during simulations. These simulations emphasise growing from a single tile to arbitrarily large structures. However, as discussed earlier, accuracy loosely requires $K/N_S \gg 1$ and so K should be chosen with the final genotype in mind. This means that earlier assemblies will be unnecessarily thorough at the expense of speed. Since the assembly graph classification scheme does not have any fixed parameters, it scales automatically alongside the evolving genotypes. These benefits become even more apparent as the polyomino model finds applications in modelling larger and more complex evolutions.

Algorithmic scaling

Assigning asymptotic computational complexity, with big O notation $\mathcal{O}$, to the two approaches is somewhat futile due the inherent heterogeneity of the genotype-phenotype mapping process. Analysing each interaction contribution in isolation is meaningless, as interactions are almost always interwoven, generating crosslinking effects. For example, doubling the number of edges in an assembly graph can double the number of SIFs to trim, make a single SIF branching point, turn all SIFs into cycles, etc. Despite the variance, high level approximations of algorithmic scaling can still be quite revealing.

In stochastic assembly, the majority of time is spent growing the assembly, scaling approximately as $\mathcal{O}\left(N_S^2\right)$. Likewise, any operations comparing two polyominoes for determinism will also be roughly $\mathcal{O}\left(N_S^2\right)$. This scaling originates from each subunit contributing to the polyomino's extent in both dimensions, and hence the quadratic time to build or compare polyominoes.

Many components of the assembly graph scheme are modified existing graph theory algorithms, and thus have relatively well-known scaling. SIF pruning checks each subunit for removal, before recursing and rechecking for $\sim \sum N_S$ checks, hence scaling as $\mathcal{O}\left(N_S^2\right)$. On the other hand, cycle detection is based on finding minimum spanning trees*, which generally scales with the number of edges E as $\mathcal{O}\left(E \log E\right)$. Other stages of classification might depend on both the number of subunits and edges.

Assembly graph classification time scaled approximately as $\mathcal{O}\left(N_S^{1.4}\right)$ when averaged over randomly generated genotypes. This reduction is significant, particularly for genotypes encoding many subunits. Certain genotypes can be classified more swiftly through stochastic assembly over assembly graphs, e.g. strong nondeterminism that is caught

*Prim's or Kruskal's algorithms are the most common for determining minimum spanning trees (MST).

quickly in assembly versus the final classification scheme step. Overall, however, assembly graphs and associated analysis offer a superior approach for quickly classifying general genotypes.

As mentioned earlier, relationships between N_S, E, and polyomino size are dependent on geometric specifics and interplay, so a simple estimate based on N_S alone is more meaningful. Importantly, through considering the partial scaling of algorithmic steps, the worst-case scaling of assembly graphs appears to be comparable to stochastic assembly with $\mathcal{O}\left(N_S^2\right)$. This supports the observed speed results, where certain genotypes benefit less from this analysis but never worse.

2.4.4 Implementation invariance

Self-assembly, as sketched out earlier, can be implemented with multiple possible algorithms. This work is based on an algorithm which considers all possible interactions concurrently, and selects from them with equal probability. This implementation matches that used in a previous study from 2014 [21]. However, earlier polyomino studies [19, 20] focused on picking random subunits and random locations and seeing if the aligned binding sites interacted. A 2016 study [23] described the algorithm as growth by random subunits, implying the same specification as the earlier papers, but references the 2014 algorithm.

Implementation choice leads to subtle differences that mostly manifest in different classification speed. In the "truth" limit, where $K \to \infty$, any choice of valid implementation will necessarily give identical classifications for any assembly. With a smaller K value, however, different implementations can have nuances leading to different error rates for unbound or nondeterministic assemblies.

A simple example is the genotype $\{(1,2,0,0),(1,0,0,0)\}$. With relaxed restrictions on mismatched binding sites, the error rate for misclassification ranges from $(7/8)^{K/2}$ to $(19/27)^{K/2}$ for the former and latter implementations respectively, matching cases b) and c) in Figure 2.11. Clearly the error rates go to zero for both implementations as $K \to \infty$, but the former implementation misclassifies nearly 10 times as many genotypes for $K = 10$. Other examples show the inverse, where the latter implementation is less accurate, so no single implementation is globally better.

As recurringly noted, there may be specific contexts where one implementation algorithm is more physically realistic and thus the appropriate choice. In general though, the choice of implementation is unintentional and potentially biases error rates unhelpfully. Removing this axiomatic dependence on implementation and the external parameter dependence on K is highly satisfactory and increases accuracy and speed across the board.

2.5 Additional benefits of assembly graph formalism

Shifting from thinking about genotypes to assembly graphs as the information carrier has several benefits. For example, assembly graphs provide information about interactions,

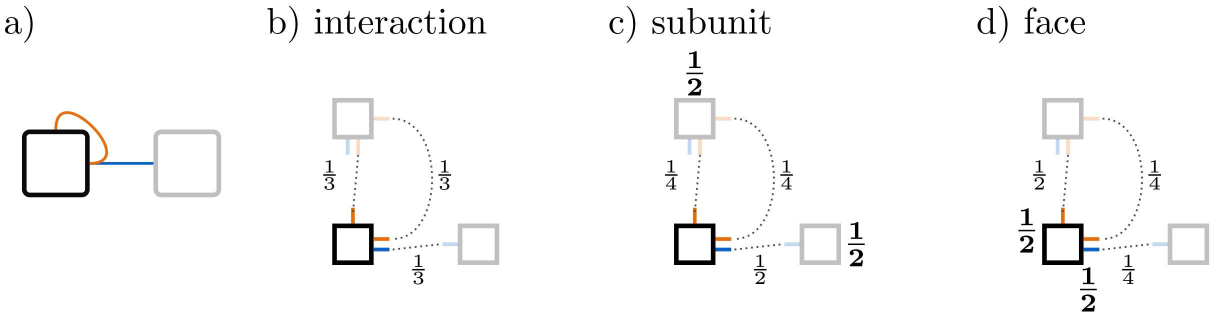


Figure 2.11: Detection of nondeterminism or unboundedness can depend on the underlying assembly implementation in a non-general way. This example assembly graph in a) can use the black subunit as a fixed seed to best demonstrate this effect. The three possible interactions do not depend on implementation, but their likelihood of being chosen during assembly does. In b), all interactions are equally likely, while in c) and d) either each new subunit or existing face is equally likely to be chosen. For the latter two, this biases the interaction usage, as observed by the different probabilities. Colour notation follows Figure 1.4.

regardless of the exact form the binding sites or interaction rules may take, allowing extremely general representations. In addition, assorted results previously derived empirically can be explained with mathematical insight. There are several concepts that the assembly graph formalism introduce or strikingly expand beyond genotypes, which have far-reaching impact on the tractability of polyomino models.

2.5.1 Assembly graph isomorphisms

Genotype-phenotype maps revolve around neutral components, a collection of genotypes connected by point mutations which correspond to the same phenotype. Previously, these components had to be discovered by stochastic assembly and grouping resultant phenotypes. Two distinct genotypes may encode the same interactions, and thus necessarily produce the same phenotype, but this was typically only known *a posteriori*. Identification of neutral components can then be done at a lower level with assembly graphs.

Since assembly graphs can be represented as graphs, two assembly graphs are the same if their corresponding graphs are isomorphic. Isomorphisms are most commonly defined for simple graphs, but have been generalised to cover other graph variants. Additional attributes such as directed and weighted edges present in the edge-labelled pseudographs have bijective mappings [34], and so isomorphisms can be applied to all aspects of assembly graphs. Most standard graph theory software packages already contain these generalisations and can be used out of the box. Some example isomorphic assembly graphs are shown in Figure 2.12. Detecting which genotypes are equivalent through sequence or subunits is virtually impossible; detecting neutral genotypes through graph isomorphisms is trivial.

Graph isomorphisms remain a current area of development, potentially belonging to the NP-intermediate complexity class. Recent advances state quasi-polynomial time can be achieved [35], and in general can be computed quickly. As such, it can be quicker to construct neutral components through isomorphisms, rather than grouping genotypes post-assembly. This approach is explored in more depth below.

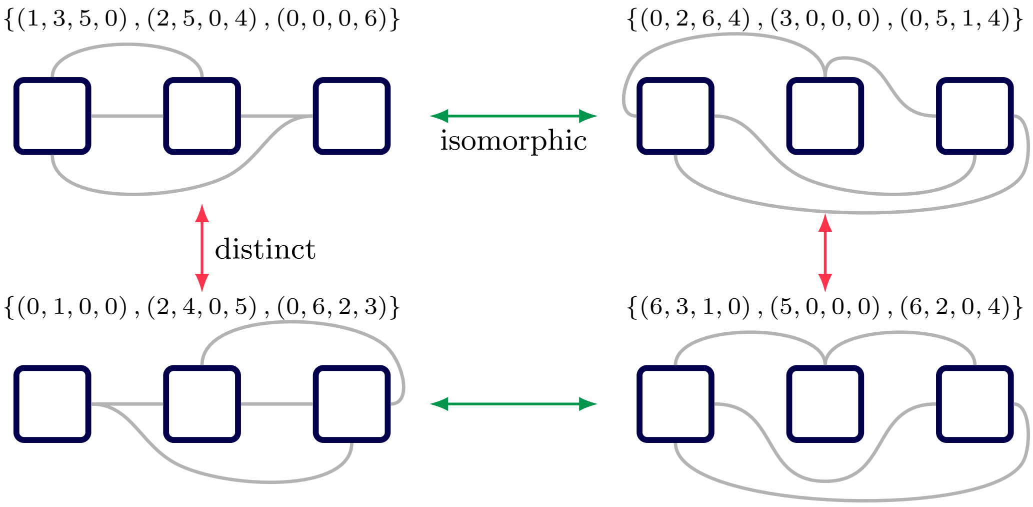


Figure 2.12: Genotypes can have extremely disparate sequences and look like distinct assembly graphs but ultimately be isomorphic and necessarily correspond to the same phenotype. Here, assembly graphs appear similar within vertical columns, but are actually isomorphic within the horizontal rows.

Chirality

The mathematical structure of graph isomorphisms also provides an unintended benefit. Most polyominoes contain some form of rotational symmetry, but fewer have reflection symmetry. This leads to many polyominoes having distinct chiral counterparts, e.g. the "left-handed" and "right-handed" Z-shaped tetramers. If free polyominoes are more relevant than one-sided polyominoes to the system under study, then identifying neutral genotypes becomes more challenging as assembly graphs are encoded with a clockwise convention, making them innately directional.

Reversal of the directed edges of the internal subunit structure corresponds to a reflection of the assembled structures. Again such an operation is well-known in graph theory as the transpose or reverse of a directed graph (interactions in assembly graphs are bi-directional and unaffected), as highlighted in Figure 2.13. This in turn is equivalent to reversing the order of the genotype, and so is not exclusive to assembly graphs. However, the language of assembly graphs naturally illuminate this insight, whereas reversing the genotype to account for reflection symmetry is not as immediately obvious.

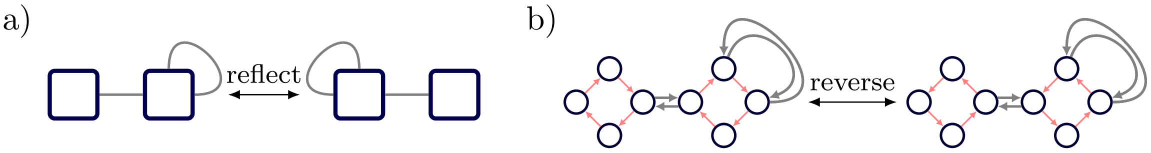


Figure 2.13: Chiral counterpart phenotypes correspond to reflections of their assembly graphs (a). Reflecting assembly graphs is equivalent to reversing the internal edges in their graph representation (b), now using a counter-clockwise convention.

2.5.2 Genotype to assembly graph mapping

Similar to most genotype-phenotype maps, which contain many more genotypes than phenotypes, there are many more genotypes than assembly graphs. Understanding how many genotypes produce the same phenotype, i.e. are in the same neutral set, can be highly valuable for the study of evolutionary dynamics and related processes. Quantifying how many genotypes map to a given assembly graph is thus an important intermediate, and can be calculated through various assembly graph symmetries and combinatorics.

There are several specific symmetries present in constructing assembly graphs from integer-labelled genotypes. For example, swapping all **1** and **2** interface types clearly will not alter assembly behaviour or change the assembly graph but is a distinct genotype. Similarly swapping all **1**s with **3**s and **2**s with **4**s will not change the interactions but yields a new genotype. The total number of these interaction relabellings possible for $N_{\text{I-pairs}}$ interacting pairs given C_I interacting types is a form of enumerative combinatorics, and is given by:

$$IR = \prod_{n=0}^{N_{\text{I-pairs}}} (C_I - 2n) = \frac{C_I!!}{(C_I - 2N_{\text{I-pairs}})\,!!}$$

In addition to interaction relabelling, the non-interacting faces can be relabelled and arranged differently, which are also combinatorial problems. This contribution to the neutral space for an assembly graph with F noninteracting faces, C_N neutral interface types, and $P = C_I - 2N_{\text{I-pairs}}$ unused interaction types (total interface types is thus $C = C_N + C_I$) is then:

$$\underbrace{\sum_{U=0}^{F} \binom{F}{U} C_N^{F-U}}_{\text{neutral}} \cdot \sum_{U'=0}^{U} \Bigg[\underbrace{\Big[\sum_{D_1}^{U'} \cdots \sum_{D_L}^{D_{L-1}} \prod_{i=1}^{L} D_i\Big]}_{\text{repeated non-interacting}} \cdot \overbrace{\prod_{U''=0}^{U'} (P - 2U'')}^{\text{unique non-interacting}} \Bigg]$$

where $L = U' - U$ and the ellipsis in the repeated non-interacting term indicates there are a total of $L - 1$ sums to consider. Combining the two expressions gives the total number of isomorphisms for a specific assembly graphs in a single orientation. Additional contributions to the neutral space of a genotype arise from general underlying assembly graph symmetries, including subunit rotations, a factor of 4^{N_s}, and subunit reorderings, a factor of $N_s!$.

However, these contributions are not all independent. For example, relabelling the interacting pair in $\{(1, 0, 0, 0), (2, 0, 0, 0)\}$ and then swapping the order of the first and second subunit leaves an identical genotype to the starting reference. Simply multiplying all the listed factors above would then over-estimate the number of genotypes mapping to that assembly graph.

No general form could be found which resolves these genotype symmetry related conflicts. Rotations and reorderings only depend on N_s, offering a mitigation to this issue.

A corrective factor which encapsulates the conflicting over-estimation can be empirically derived by comparing the analytic form and the observed cardinality of the neutral set. This corrective factor is constant, and so can be calculated from the minimal relevant genotype space, and then applied to any number of interface types. An example genotype-phenotype mapping for two subunits is shown in Figure 2.14.

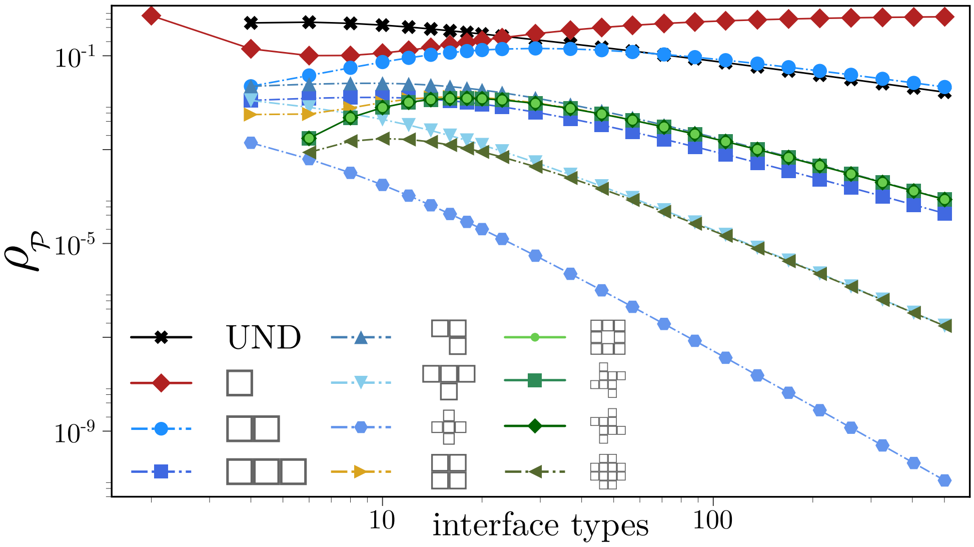


Figure 2.14: Fractional phenotype abundance, $\rho_{\mathcal{P}}$, for all twelve phenotypes possible in a two-subunit genotype. Abundance changes significantly with the number of allowed interface types, to the extent that the rankings can change. For few interface types, fractional abundance is dominated by configurational combinatorics. In the larger limit, it is mainly fixed by the number of interacting pairs and non-interacting faces, causing the observed collapses.

Note the calculations introduced above find the size of the neutral *set*, which is all neutral genotypes. They do not provide information on neutral *components*, which are connected components of the neutral set in the network of point mutations. These two quantities can be the same for high-dimensional genotypes spaces, where two strongly different genotypes can be connected by a long sequence of neutral point mutations. However, neutral sets generally break into multiple smaller neutral components, as observed for multiple polyomino genotype-phenotype maps [21, 36].

2.5.3 Efficient genotype space sampling

While many genotypes map to the same assembly graph, many assembly graphs uniquely map to phenotypes. Therefore enumerating assembly graphs allows near perfectly efficient sampling of a phenotype space. Unfortunately, due to the multiple types of symmetry present in genotypes, the exact enumeration of assembly graphs remains challenging. There are, however, multiple intermediate approaches which compress genotype space size while preserving the assembly graph and phenotype spaces. This compression allows analysis of significantly larger genotype-phenotype maps than were conventionally possible.

The simplest step is producing subunits which remove meaningless cyclic symmetry;

these sequences are known as *necklaces** and have previously appeared in polyomino models [19]. Necklaces still do not account for interaction relabelling or subunit ordering.

The first element of a genotype can either be neutral or possibly interacting, and so should only take a value of **0** or **1** respectively. Taking any other value will necessarily be degenerate, as they will be isomorphic to an assembly graph starting with the above options. Only once an interface type of **1** exists in the genotype should future elements take values of **2** (interacting partner) or **3** (new possible interaction). This "unlocking" of interaction types continues in this fashion, substantially cutting down on generated genotypes.

Necklaces and interaction unlocking can be combined into a single generator. Recursive necklace algorithms, such as the FKM CAT (constant amortized time) algorithm [37], can be modified in this way, only expanding the genotype space as necessary. The only input required is the maximum value of interface type allowed. This algorithm is highly efficient in both time and memory, and can quickly generate the entire set of minimal subunits for a genotype space.

These minimal subunits can be combined to form minimal genotypes. Additional rejection criteria can be used given a fixed genotype size, such as ignoring any genotype which contains any non-zero interface types which are not involved in interactions. At this stage, the genotype space is generally compressed by several orders of magnitude into the minimal genotypes. Isomorphism tests can winnow out any surviving degenerate genotypes to produce unique assembly graph topologies. The scale of this compression is shown in Table 2.1.

Table 2.1: Integer-label polyomino systems can be represented as $S_{N_S,C}$, for the number of subunits and interface types. The genotype spaces are strongly degenerate, and symmetries can be stripped to form necklaces, minimal genotypes, and assembly graph topologies. The compression factor (CF) measures the reduction factor between the genotype space and topology space, and scales strongly as larger systems are generally more redundant. The $S_{4,16}$ case was previously studied [21], but resorted to sampling the genotype space.

	Space size					
System	Genotypes	Necklaces	Minimals	Topologies	Phenotypes	**CF**
$S_{2,8}$	$\sim 2 \times 10^7$	$\sim 1 \times 10^6$	62	41	12	$\sim 10^5$
$S_{3,10}$	10^{12}	$\sim 2 \times 10^{10}$	21054	9838	148	$\sim 10^8$
$S_{4,16}$	$\sim 2 \times 10^{19}$	$\sim 7 \times 10^{16}$	$\sim 2 \times 10^7$	$\sim 3 \times 10^6$	3887	$\sim 10^{13}$

With the substantial compression achieved, the $S_{4,16}$ phenotype space can be exhaustively constructed. Although there are minor model variations in terms of fixed versus random seed, it is highly likely the original study [21] on this system underestimated the quantity of phenotypes. Extremely scarce phenotypes are unlikely to be found by randomly sampling genotype space. Assuming similar model comparison ratios for the $S_{3,10}$ system, the sampling method potentially missed 30% to 50% of phenotypes in $S_{4,16}$. Even sampling

*Such sequences are called *bracelets* if reflection symmetry is also removed.

topology space gives a substantial improvement, as topologies and phenotypes are much closer to a one-to-one correspondence.

This methodology has been adapted in ongoing work by Jouffrey, Leonard, and Ahnert for extending genotype-phenotype maps to previously incalculable spaces. Map properties can be estimated by randomly reintroducing neutral mutations to each reference genotype in the unique assembly graph topology space. Sampling over a large enough quantity of these mutated reference genotypes provides a diverse but balanced estimate of the entire space.

2.6 Conclusion

(Un)boundedness and (non)determinism are fundamental properties of self-assembling systems. The assembly graph classification scheme has been refined since first introduced [25], rectifying several key flaws in the cycle analysis and SIF treelike arguments. This framework also enables a more conceptual understanding of proteopathy, where minor mutations to the assembly graph of a deterministic complex can quickly induce uncontrolled growth.

The classification of assembly behaviour through assembly graphs is also faster and more accurate compared to conventional methods of stochastic "consensus". Issues regarding the optimal trade-off of accuracy and speed are resolved, particularly in evolutionary simulations. Axiomatic dependence on the assembly implementation algorithm is similarly removed.

The strictness of the classification scheme can be reduced through extensions such as seed dependence and variable interaction environments. Assembly graphs, with their mathematical fabric, also naturally allow isomorphic genotypes to be detected and genotype space to be compressed. Previous analyses thought only possible through sampling can now be tackled exhaustively, e.g. a four-subunit genotype-phenotype map. In general, assembly graphs provide an organic language for discussing self-assembly and understanding the nontrivial relationships between interactions and assembled structures.

2.6.1 Universal computing

In computability theory, the *halting problem* states that it is not possible in general to determine if an algorithm[*] will terminate or run forever. Amongst other contemporaries, Turing first showed such a result in 1936 [38], and has since been scrutinised and generalised without fault.

The polyomino model used in this work is a highly specialised form of a much more general class of tile assembly models (TAMs), as introduced in Chapter 1. These TAMs can take on diverse forms, and are of considerable interest to the fields of mathematical logic and theoretical computer science. Crucially, some TAMs have been shown to be

[*] More formally a Turing machine.

Turing complete (also known as computationally universal), meaning they can emulate any algorithm with the assembly of subunits. As such, determining the boundedness of any such assembling structure *a priori* would be equivalent to overthrowing the halting problem.

Fortunately, no such claim is made here. Interactions are explicitly defined in this work as non-cooperative, meaning they do not depend on other interactions occurring simultaneously. This class of TAMs has been shown to be not Turing complete [39], and requires substantial modifications not compatible with the presented assembly graph analysis to be capable of universal computing [40]. While this means that our analysis cannot be fully generalised to more potent classes of TAMs, it is reassuring that there is no conflict with the Halting problem.

CHAPTER

THREE

BINDING STRENGTH DYNAMICS

Key messages

- Binary string binding sites substantially generalise the polyomino model.
- Nondeterministic assembly selection pressure drives interaction strength evolution away from equilibrium.
- Optimal interaction strength ratios can retrospectively infer likely evolution pathways.

This chapter is based on the publication *Evolution of interface binding strengths in simplified model of protein quaternary structure* by Leonard and Ahnert [41]. All results are independent and novel work by Leonard under the supervision of Ahnert.

3.1 Introduction

Although the standard polyomino model has proven useful in diverse contexts, the integer interface types are rather artificial. The number of interface types can strongly affect phenotype abundances and evolutionary dynamics, as seen previously in Figure 2.14. Picking a configuration space with 10 interface types compared to 20 has little meaningful translation to biological systems, and so the impact on observations is impossible to contextualise. Likewise, having interface types only interact in pairs is limiting and not something observed in natural self-assembly.

Generalising binding site labels with binary strings is a better coarse-grained approximation for protein-protein interfaces, and can mitigate the aforementioned issues as well as enabling the study of a wider range of self-assembly dynamics. Protein binding models using binary [42] or otherwise simplified residues [43] demonstrate general properties that agree well with experimental evidence. Such a simple binding site description will never capture the nuance of amino acids (e.g. aromatic, aliphatic, acyclic, etc.), but can capture suitable detail while retaining analytic potential.

Although difficult to measure directly, binding affinity, i.e. interaction strength, is central to understanding assembly dynamics [44]. Binding affinity has been observed to substantially change through mutations to polar or charged groups [45]. This suggests that a coarse-grained model with limited states (0/1 states approximating e.g. polar/nonpolar) may be sufficient to capture broad properties of the evolution of interaction strength.

In addition to measuring changes in interaction strengths, the generalised model can also explore the relationship between binding strengths and evolutionary pathways. Several recent studies have revealed connections between evolutionary pathways and assembly properties such as stoichiometry [46], symmetry [47], interaction topology [10], and binding strengths [48]. Recreating these observations in a simple model may assist in generating a high-level perspective on how assembly dynamics are reflected in evolutionary histories.

3.2 Generalised lattice self-assembly model

To generalise the binding site labels, single integers are replaced with binary strings: sequences of 0's and 1's. Binary strings of length L have 2^L unique configurations, hence longer binding sites correspond to more possible genotypes. A genotype encoding N_S subunits is then a long binary sequence of length $4N_SL$.

Two adjoining binding sites will have a "head-to-tail" alignment, and the strength of their interaction is related to how many bits complement their opposite entry. Formally, this is the Hamming distance between one binding site and the reverse of another. The interaction strength can then be written as:

$$\hat{S} = \left(I_1 \oplus I_2^\top\right)/L$$

where the $\top$ indicates reversing a binary string. Interaction strength is normalised by the binding site length L such that $\hat{S} \in [0, 1]$. Weak interactions do not successfully bind, as can be defined with a critical interaction strength, such that interactions only form if $\hat{S} \geq \hat{S}_c$. An example of the generalised binding site model is shown in Figure 3.1.

Another feature inherently introduced alongside interaction strength is binding precedence. Assemblies can preferentially select stronger interactions to bind over weaker ones, adding some thermodynamic pragmatism. Explicitly, interaction strength can map to a binding probability, for example through the relatively simple form:

$$p_\mathrm{b}\left(\hat{S}\right) = \begin{cases} \hat{S}^T & \text{if } \hat{S} \geq \hat{S}_c \\ 0 & \text{otherwise} \end{cases}$$

Where $T \in [0, \infty)$ is an abstract temperature parameter. This parameter enables a tunable bias during assembly towards stronger interactions. The expected number of binding attempts an interaction will take is the reciprocal of the probability, and is

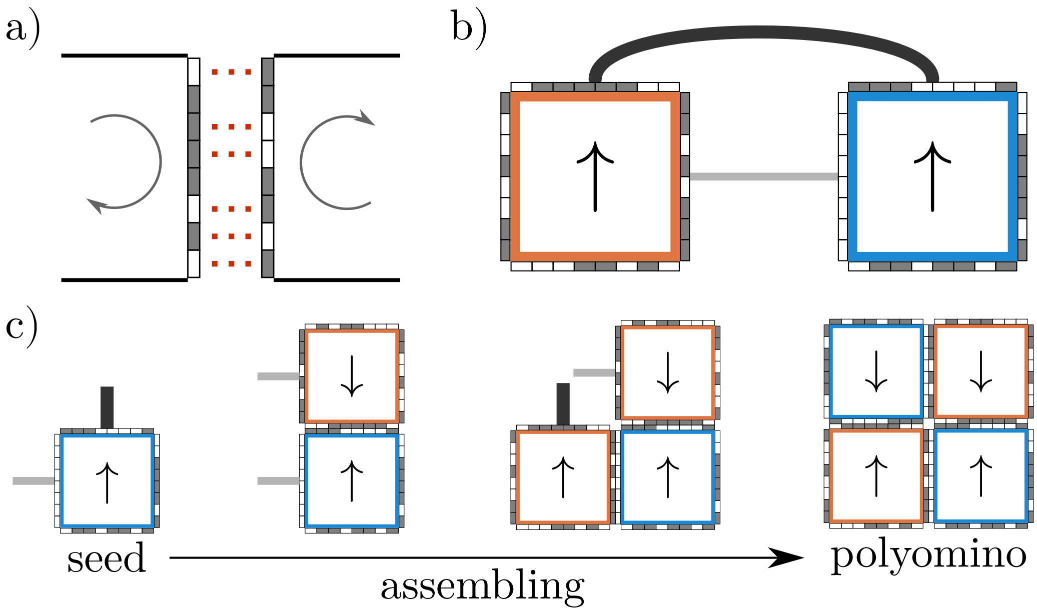


Figure 3.1: a) Complementary bits contribute positively to binding strength, indicated by dotted lines, on two counter-aligned binding sites. b) A detailed example assembly graph contains two interactions of different strengths: $\hat{S} = 0.875$ (black) and $\hat{S} = 0.75$ (grey). Interaction width reflects strength. c) Assembly proceeds as standard but now with weighted probabilities, stochastically growing from a random initialisation to a complete structure. Colourings are as described in Figure 1.4.

effectively the expected binding time for that interaction. For choices of $T > 0$, stronger interactions assembly more quickly than weaker ones on average. More physical forms of binding probability were also examined with similar qualitative results, like using exponential Boltzmann factors. However, with the computationally-limited length of binding sites, the above form provided the most clear dynamics over a wider range of strengths.

Notably, interactions within the generalised model form a proper superset of integer-labelled interactions. By taking $\hat{S}_c = 1$ and picking $L \sim \log_2 C$, there exists a nearly bijective mapping of labels. For example, two interacting binding sites given $L = 4$ are 0001 and 1110, which map to allowing an interaction between integer interface types **1** and **14**. Therefore any results derived in the integer-labelled model can trivially be recovered in the generalised polyomino model.

Self-interactions also naturally exist within generalised binding sites, appearing without special cases or further rules. In all existing implementations of the integer-labelled model, one interface type interacts with only one other interface type. This could be relaxed in principle, but would likely lead to ad hoc interaction choices.

Previously, self-interacting interfaces were forced into the model using negative integers [21], whereas now there is a singular form for all interactions. Self-interactions are explicitly discounted in this analysis, but form the foundation of Chapter 4.

3.2.1 Interaction properties

Despite the gain in qualitative realism, the self-assembly model with generalised binding sites remains largely tractable. Many of the generalised properties are without analogue in the integer-labelled model, and hence allow for modelling completely new dynamics. Several of the analytic results can also provide deeper understanding and context to the underlying dynamics governing biological systems.

Transitivity

One of the most immediate differences to the integer-labelled model is the relaxation on interaction transitivity. Previously, possessing information on what a binding site interacted with was sufficient to determine all possible interactions regarding either interface type. With the notation of '↮' representing interacting partners, integer interface types A, B, C obey:

$$(A \leftrightarrow B) \wedge (B \leftrightarrow C) \rightarrow (A = C)$$

Again with the generalised interactions forming a proper superset, the left-hand side of the above statement is a necessary but not sufficient condition with binary string binding sites. Knowing two given binary strings interact does not contain much information on what else they may interact with. As binding site length increases, the mutual information contained by an interaction asymptotically goes to zero. Opposite to above, binary string binding sites D, E, F, G have the property that:

$$(D \leftrightarrow E) \wedge (E \leftrightarrow F) \wedge (F \leftrightarrow G) \nrightarrow (D = F) \vee (D \leftrightarrow G)$$

With less transitivity amongst interactions, more intricate assembly graphs are possible in the generalised polyomino model. Importantly, now branching points can occur between non-identical binding sites. In the integer-labelled model, branching points required frequent repeating of interface types, whereas protein complexes rarely, if ever, repeat exact binding sites. While the binary string representation is not entirely realistic, it does minimise another artificiality of the integer polyomino model. An example of a non-transitive interaction, even with short binding site length, is shown in Figure 3.2.

Interaction radius

As demonstrated above, any given binary string can likely interact with multiple others. For two arbitrary strings, there is a fifty-fifty chance that any opposing bits will interact. Enough bits must interact to reach or exceed the threshold in order to form an interaction, which follows a binomial distribution. The number of interaction partners I_r is then the sum over all strengths at or over the threshold, otherwise known as the survival function, as:

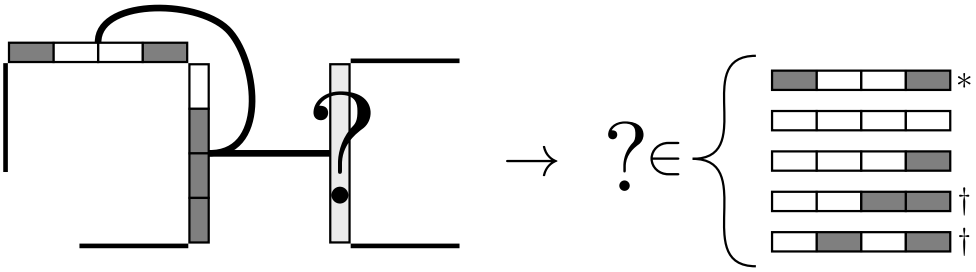


Figure 3.2: There are many configurations of binding sites which can produce the two interaction assembly graph shown. Picking the two binding sites for the left subunit would have fixed the other binding site labelled ? in the integer-labelled model due to transitivity. However, here there are 5 possible binding sites allowed even with such a small configuration space of $L = 4, \hat{S}_c = 0.75$. The first, marked by $*$, is trivially the transitive option, while the remaining four are different. The final two marked with † can self-interact, so technically add an interaction to the assembly graph if they are enabled.

$$I_r = \sum_{m=\lceil L\cdot\hat{S}_c\rceil}^{L} \binom{L}{m}$$

which is equivalent to considering how many ways there are to distribute mutations while maintaining the interaction strength. Crucially, the number of interacting partners grows rapidly as L increases, $\hat{S}_c$ decreases, or both. A similar result can be found for the likelihood that a random genotype will not have formed any interactions. This proceeds with the binomial cumulative distribution function for $n = 4N_S$ binding sites as:

$$p(E=0) = \left[2^{-L} \sum_{m=0}^{\lceil L\cdot\hat{S}_c\rceil -1} \binom{L}{m}\right]^{n(n-1)/2} = \left[1 - \frac{I_r}{2^L}\right]^{n(n-1)/2}$$

Even initialising two subunits without any interactions is difficult for shorter binding site lengths. Although increasing binding site length increases the number of possibly interacting sequences, the proportion relative to all sequences decreases*. This proportional interaction "radius" shrinking with increasing length or threshold makes it more difficult for interactions to form. In the extreme limit of long string length and high strength threshold, the radius is effectively zero and any two random binding sites are unlikely to interact. Length and threshold should be chosen appropriately for the dynamics under examination, such that interactions do not appear constantly but also can be eventually formed by evolution. An example of the nontrivial effect of these parameters on interaction radius is seen in Figure 3.3.

Attempting to translate L and $\hat{S}_c$ to any biological meaning should be done cautiously, but the balance between forming interactions too easily and too infrequently could be of interest in modelling the co-evolution of binding site size. In densely "crowded" environments, e.g. outside a cell nucleus, proteins likely require greater binding specificity to avoid agglomerating, which might correspond to larger binding regions. Such behaviour

*Technically only if $\hat{S}_c > 0.5$, which is the case in all presented results.

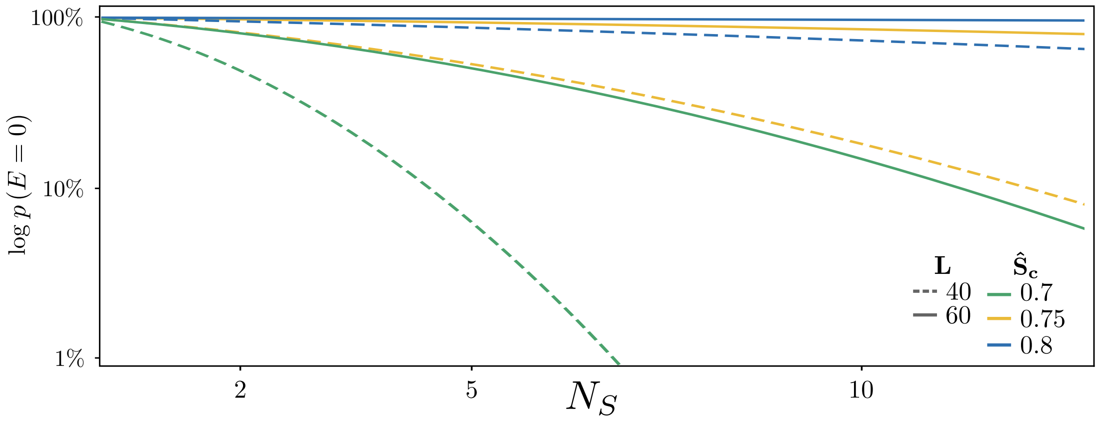


Figure 3.3: The probability of forming zero interactions, $p\,(E = 0)$, quickly decays for short binary strings or low strength thresholds. For long binary strings or high strength thresholds, even genotypes with many random subunits will not form any interactions. Constant changes in parameters do not result in constant changes to radii, seen by both $\hat{S}_c = 0.7$ lines below $\hat{S}_c = 0.75$, but the same is not true for $\hat{S}_c = 0.75$ and $\hat{S}_c = 0.8$.

has been observed in the literature for multiple contexts [49–51]. Some ideas to more generally model these dynamics are expanded on further in Chapter 5.

3.2.2 Weakly nondeterministic assemblies

Uncontrolled nondeterminism is certainly biologically unfavourable, amounting to random formation of a protein complex. However, it is not atypical for proteins to occasionally misassemble and then be "recycled" by proteasomes [52, 53]. This behaviour can be thought of as *weak* nondeterminism, where the assembly is performed correctly most of the time but some pathways lead to errors. As such, relaxing the polyomino model to allow for weak nondeterminism is critical to understanding more complex protein assembly behaviour.

Similarly, the steric growth constraint employed in the previous chapter is necessary to ensure guaranteed assembly of the same structure. However, this is also relaxed to model more general proteins, which can incorporate some aspects of self-limitation or spatial hindrance to self-assembly [54].

Nondeterministic assembly is still less optimal, and so might still incur some form of penalty relative to deterministic assembly. This can be introduced via a nondeterminism penalty parameter $\gamma \in [1, \infty)$, which lowers the evolutionary success achieved by how unreliable the assembly was.

3.2.3 Strength evolution as a Markov process

In order to understand the evolutionary dynamics acting on interaction strengths, it is imperative to understand how strength changes in the absence of selection pressures. Interaction strengths can be modelled as states in a Markov process, where mutations step across these states. With only a few assumptions, an analytic form for the baseline evolution of strength is easily achievable.

The key assumptions are:

- Infinite population size.
- Single point mutations.
- Dropping below critical strength is fatal.
- Stronger interactions have no direct evolutionary advantage.

Standard parameterisation of the evolution simulations approximately satisfy all these assumptions, and so the Markov process estimation provides a meaningful reference. The last assumption is more subtle. Since selection pressure acts on a phenotype rather than the underlying assembly, it is realistic that as long as a phenotype is achieved there is no additional direct reward for its interaction strengths.

Formally, the full transition matrix requires a transition weight to the fatal state immediately below the strength threshold, and weights to "refill" the fixed-size population in other states. This would mean the matrix itself would be changing at every iteration as those state populations changed. Under the approximation that all valid strength states are equally fit, selection dynamics proportionally redistribute the fatal transitions. Ultimately, the redistribution dynamics and transition below the threshold can be dropped without loss of generality as they only affect a normalisation factor that is discarded.

As mutations only allow single steps to adjacent states above or below, the transition matrix $\mathbf{Q}$ only contains non-zero elements in the super- and subdiagonals. Transition probabilities to move up or down are directly related to the strength state and complementary weakness as $W_i = i/L$ and $W_i^\dagger = 1 - i/L$ respectively. Since strengths have to be at or above the threshold to matter, the lowest allowed state is given by $c = \left\lceil L \cdot \hat{S}_c \right\rceil$. The resulting Markov process for the simplified strength mutation dynamics is shown in Figure 3.4.

The number of states in a chain, i.e. the number of valid interaction strengths, is given by $N = \left\lfloor L \cdot \left(1 - \hat{S}_c\right) \right\rfloor + 1$, connected by $N - 1$ steps. The general transition matrix is thus:

$$\mathbf{Q} = \underbrace{\begin{pmatrix} 0 & W_{c+1} & & \\ W_c^\dagger & 0 & \ddots & \\ & \ddots & \ddots & W_L \\ & & W_{L-1}^\dagger & 0 \end{pmatrix}}_{N}$$

Although the strength state transition matrix is not strictly positive, it is non-negative and the resulting directed graph is strongly connected: all states are reachable from all other states via some path. This matrix therefore satisfies the Perron-Frobenius theorem, which

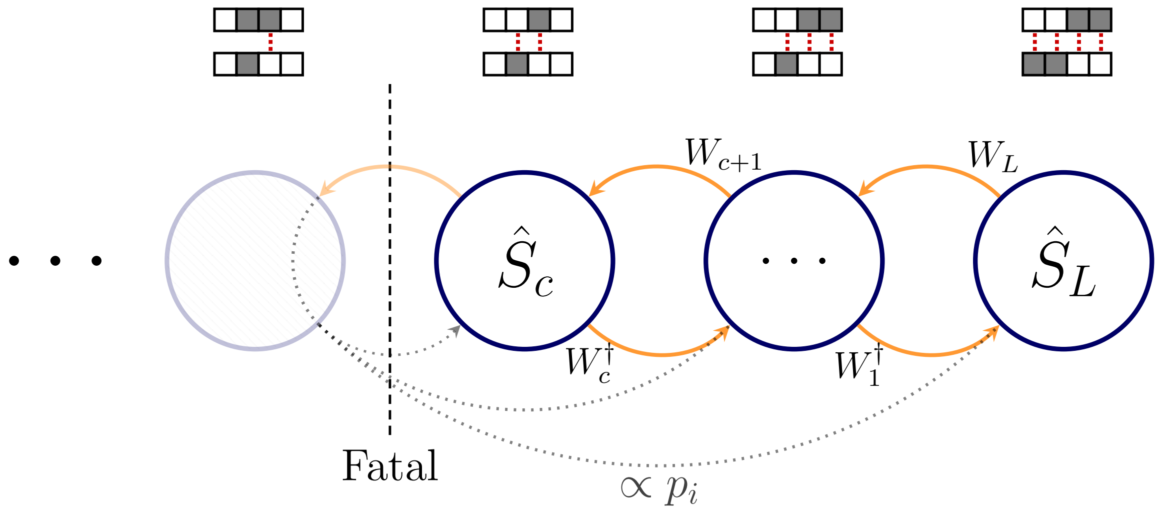


Figure 3.4: The simplified Markov chain only involves strength states at or above the threshold, as states below and the repopulating selection dynamics (faded) can be ignored. An example $L = 4, \hat{S}_c = 0.25$ chain is shown here with possible binding sites, where the transition weights are $W_{c+1} = 0.75, W_L = 1, W_c^\dagger = 0.5, W_1^\dagger = 0.25$.

states there exists an all-positive eigenvector $\underline{\pi}_{\text{PF}}$ which corresponds to the steady-state distribution. The steady-state expectation of binding strengths is then:

$$\langle \hat{S} \rangle = \underline{\pi}_{\text{PF}} \cdot \underline{\hat{S}}$$

Where $\underline{\hat{S}} = \left[\hat{S}_i, \cdots, \hat{S}_k\right]^\intercal$ is a vector corresponding to the valid strengths. Transient expectations are slightly more complicated, and depend on the rate of mutations μ' (adjusting the mutation probability μ only occurring to this interaction). Initially, interaction strength occupation density is entirely localised to the lowest state as $\underline{p}_0 = [1, 0, \cdots]^\intercal$. After g generations, the expected strength is:

$$\langle \hat{S} \rangle_g = \left(\mu' \mathbf{Q} + (1 - \mu') \mathbf{I}\right)^g \underline{p}_0 \cdot \underline{\hat{S}}$$

Importantly, while the eigenvalues change between the steady-state and transient matrices, the eigenvectors are invariant under the transformation. Expressed more rigorously:

$$\lim_{g \to \infty} \left(\mu' \mathbf{Q} + (1 - \mu') \mathbf{I}\right)^g \underline{p}_0 = \underline{\pi}_{\text{PF}}$$

Interestingly, the rate of this asymptotic approach is bounded by expressions containing the second largest eigenvalue [55]. As stated above, the eigenvalues do change with the μ' parameter, as is physically expected as higher mutation rates should reach equilibrium sooner. The bounds are not especially important, but the qualitative connection between the mutation rate and how quickly a population should equilibriate is a useful reference for analysing the simulations.

3.3 Simulated evolution of binding strengths

Evolution is simulated through a minimal mutation-selection process. Individuals are haploid, meaning they carry only a single copy of their genotype, and reproduce asexually and then die at the end of every generation. Generations are therefore discrete, similar to the Wright-Fisher model of genetic drift. Evolution proceeds by mutating the genotype and then selecting for phenotypes for a fixed number of generations.

Mutations occur with a fixed probability to flip each bit in the genotype during reproduction, generally tuned so one mutation is expected per genotype per generation. Selection takes the form of a roulette-wheel, where individuals with higher fitness have a proportionally higher chance of reproducing. Fitness is an abstract measure of evolutionary potential, and previous polyomino studies [20] typically relate fitness to phenotype size. Assembly complexity can increase without necessarily producing a larger polyomino, and so a slightly more general form of fitness is required. A more algorithmic overview is provided in Appendix A.3.

Individuals are assigned fitness corresponding to their complexity as they evolve from monomers to larger structures. In this case, complexity is defined as the number of interactions, N_I, present within that individual's assembly graph. The fitness is weighted by the nondeterministic penalty γ, based on the fraction ϕ of assemblies which matched the target phenotype for the individual's assembly graph. The target phenotype for an assembly graph is defined as the polyomino expected to assembly the most frequently (see later in Figure 3.6). Less regular assembly of the target phenotype rewards a smaller portion of the polyomino fitness, with a greater reduction as γ increases. The fitness value for individual i is then:

$$f_i = \left(2^{N_I}\right) \cdot \phi_i^{\gamma}$$

Fundamentally, fitness rewards more edges in an assembly graph and penalises inconsistent assembly. The additional complications in the fitness equation are designed such that fitness ratios are proportionally the same across transitions. They were chosen to ensure economical simulations and emphasise the different aspects of binding strength evolution. Further details on these parameters and others used in these simulations are given in Table A.1.

Phylogenetic tracking is straightforward under this framework of asexual evolution, since any newly formed interactions and phenotype transitions can be traced directly to unique mutation events. All descendants of these individuals can then be tracked, recording population-averaged interaction strengths from their evolutionary histories. Reconstructing entire ancestral trees allows pinpoint examination of which dynamics are in play and what contexts activate them, at any given time for all assembly graphs and phenotypes. An example evolutionary history demonstrating population-averaged properties is shown in Figure 3.5.

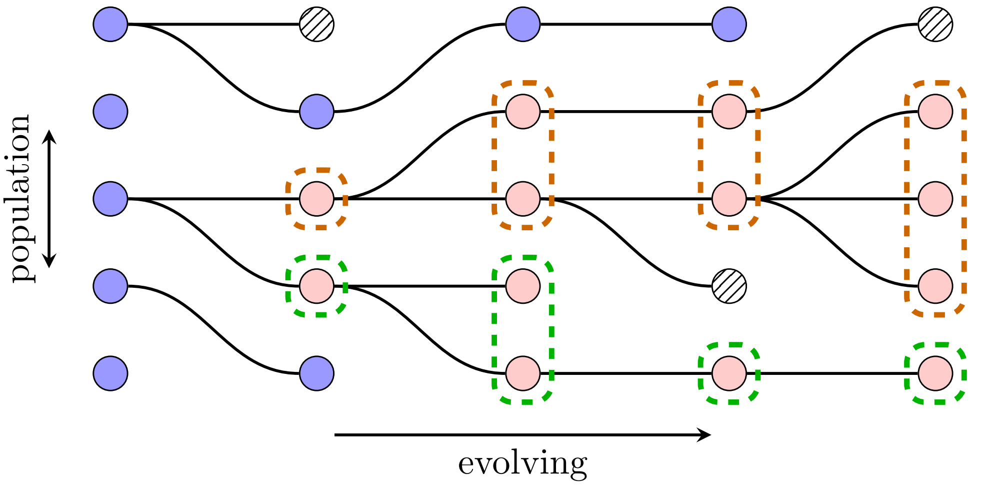


Figure 3.5: With complete information on individuals' assembly graphs and phenotypes, as well as parentage, specific properties can be tracked over the course of this example evolution. Here, red and blue circles represent different phenotypes, while hatched circles are deleterious. An arbitrary property can be tracked across, for example, the red phenotype for the two independently originating sub-populations, grouped by green or brown dashed lines.

3.3.1 Model subset of phenotypes

The following results are extremely general and have been observed in unrestricted evolution of general phenotypes. However, viewing these dynamics through the evolution of a particular fixed subset of six phenotypes provides an informative demonstration. This phenotype subset contains a range of observed behaviours, with approximately comparable evolution pathways. Each of the four more complex phenotypes can evolve from two possible ancestors via point mutations, enabling multiple pathways to reach any terminal state. The six assembly graphs and dominant phenotypes are shown in Figure 3.6.

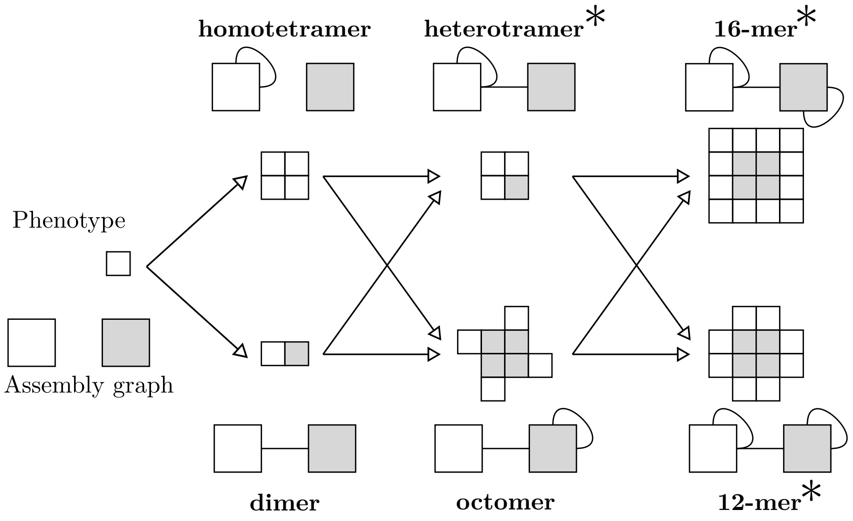


Figure 3.6: Evolution starts with a population of subunits without any interactions, i.e. monomers, which can evolve through multiple trajectories through the six assembly graphs. They are grouped vertically by number of interactions, which increases from one to two to three moving left to right. Phenotypes marked with $*$ are nondeterministic, where the displayed polyomino is the desired assembly.

Within the vertical groups of Figure 3.6, phenotype abundances are approximately equal.

The only exception is that dimers are twice as common as homotetramers. Phenotype abundances can be derived through combinatorial arguments regarding interactions or through sampling random sequences. Ancestry can depend on abundances, where a more common phenotype can evolve more often over a more fit phenotype, referred to as *arrival of the frequent* [56]. As such, evolving through the dimer is a combinatorially favoured step in any pathway to a higher complexity phenotype.

3.3.2 Interaction properties generate selection pressure

Any deviations from the appropriately justified strength expectations must therefore be driven by unaccounted additional dynamics. In order to isolate different dynamics at play, interactions are categorised by two distinct metrics: phenotype and interaction topology. Selection acts on phenotypes, and so different phenotypes may exhibit different dynamics. Interaction topology can be further subcharacterised by two properties: is the interaction inter- or intra-subunit, and are either binding sites used in other interactions (branching points) or are they exclusive.

Interaction strength evolutions for the six specified assembly graphs are shown in Figure 3.7. All interactions in the three deterministic phenotypes (top left, bottom left, bottom middle), approximately follow the transient expectations from the Markov process, delineated by the dashed line. Conversely, interactions present in the three nondeterministic phenotypes (top middle, top right, bottom right) mostly diverge from expectations. However, one interaction in the 16-mer phenotype (top right) does follow expectations, so the effect cannot be purely related to deterministic versus nondeterministic phenotypes. This particular interaction is discussed further below.

Ancestry can be directly inferred in all cases by considering the post-transition interaction strengths. New interactions typically only form at the minimum allowed strength, while older interactions have had time to equilibrate to some higher value. However, in all four cases of multi-interaction, higher complexity evolutions, interaction strength steady-states contain no trace of ancestry. Instead, the steady-states are determined only by the current phenotype's selection pressure. The time taken to reach this state depends on the mutation rate and other selection parameters, but ancestry is erased on a much shorter timescale than evolving to a further phenotype.

Some real proteins do preserve historical strengths, but this is likely due to external influences such as thermodynamic stability not considered here. Recent work suggests hydrophobic amino acids form stronger interactions but enhance aggregation risk, and require long periods of evolution to safely intersperse these potent residues [57]. As such, older, core interactions can display higher strengths.

One notable unexpected dynamic was that deterministic and nondeterministic assemblies adapted at different rates, despite an equal mutation rate. The peak observed rate of binding strength increase in the 12-mer is approximately triple the rate in deterministic assemblies. In hindsight, such an observation is fairly intuitive, as mutations which alter

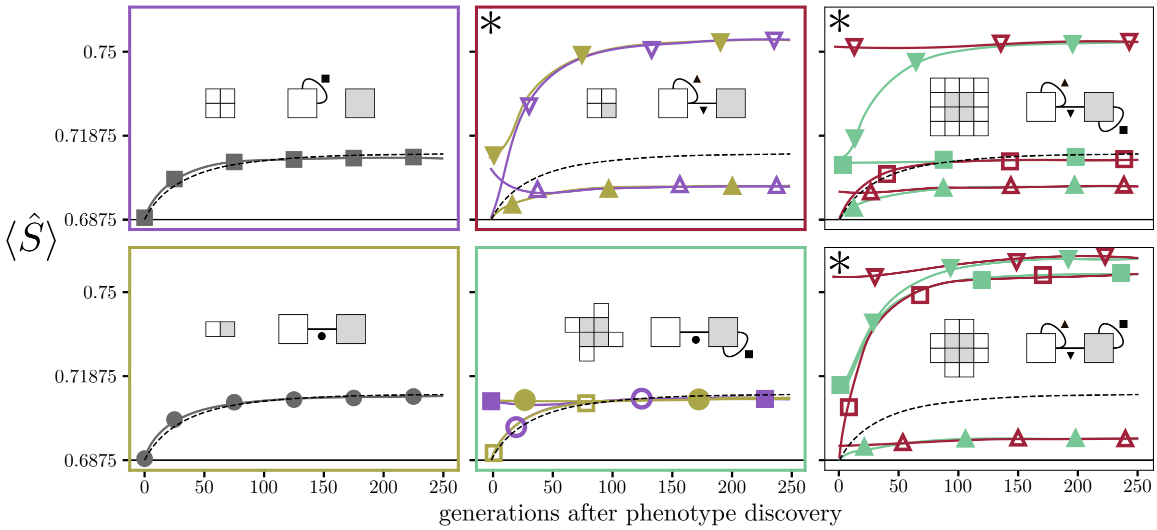


Figure 3.7: Each panel corresponds to a different phenotype's evolution, with marker shapes indicating interaction topology. Line colours of the strengths match the border colour of the immediate ancestor. The black dashed line represents the Markov chain expectations. Strengths are averaged across 10,000 simulations to stabilise the stochastic nature of evolution, and tracked for 250 generations after that phenotype was discovered. Fluctuations for individual simulations were commonly within ±0.01 of the plotted mean. Strengths for the nondeterministic panels, indicated again by $*$, diverge rapidly from expectations, while deterministic evolution is well approximated.

binding strength correctly or incorrectly are more strongly selected or purified respectively in nondeterministic assemblies. This parallels evolution in real proteins, where unstable proteins can adapt more quickly to selection pressures [58].

Steric interference and selection ordering

All three nondeterministic phenotypes are nondeterministic due to steric constraints, where multiple subunits want to bind into the same lattice site. This arises when there are multiple possible assembly pathways, and picking one ordering of assembly precludes later assembly steps from occurring. Such behaviour can be thought of as "competing" interactions. Greater determinism can achieved by minimising this competition: strengthening the preferred interaction and weakening the others. With the relationship between interaction strength and binding time, this can be reframed as assembling the "core" of the phenotype before adding periphery elements to ensure correct assembly.

Selectively strengthening or weakening interactions to maximise determinism exactly describes the dynamics observed previously in Figure 3.7. In the case of the heterotetramer (top middle), the assembly is near deterministic if the inter-subunit binding occurs before the intra-subunit bindings, i.e. stronger and weaker strengths respectively. The notion of competing interactions also resolves the mystery of the one interaction in the nondeterministic 16-mer (top right) that follows expectations. This interaction is not in competition with any other interaction and thus is not driven stronger or weaker than steady-state expectations. Regardless of the order of assembly, this interaction is never sterically blocked, and so evolves independently.

Universality

The magnitude of selection pressure on interaction strength depends on the choice of related parameters, e.g. nondeterministic penalty γ and temperature T. Clearly the maximum determinism could be achieved by fully strengthening one interaction to $\hat{S} = 1$, and weakening the other to $\hat{S} = \hat{S}_c$, but these states are unstable and cannot be maintained without equally extreme selection pressure. Instead, the competitive equilibrium is a state of "good enough", where the combination of interaction strengths minimises fitness loss due to the nondeterministic penalty.

Increasing the penalty or decreasing the temperature should promote a larger gap in strength, while higher temperature or lower penalty lessen the pressure maintaining that gap. The gap in interaction strengths relates to the differences in binding probability p_b through the temperature parameter. Provided $T > 0$ and $\gamma > 1$, where selective ordering is rewarded, some magnitude of gap should exist, and is exactly what is seen in Figure 3.8 for the heterotetramer phenotype example. As expected, as selection pressures increase so do the observed interaction strength gaps.

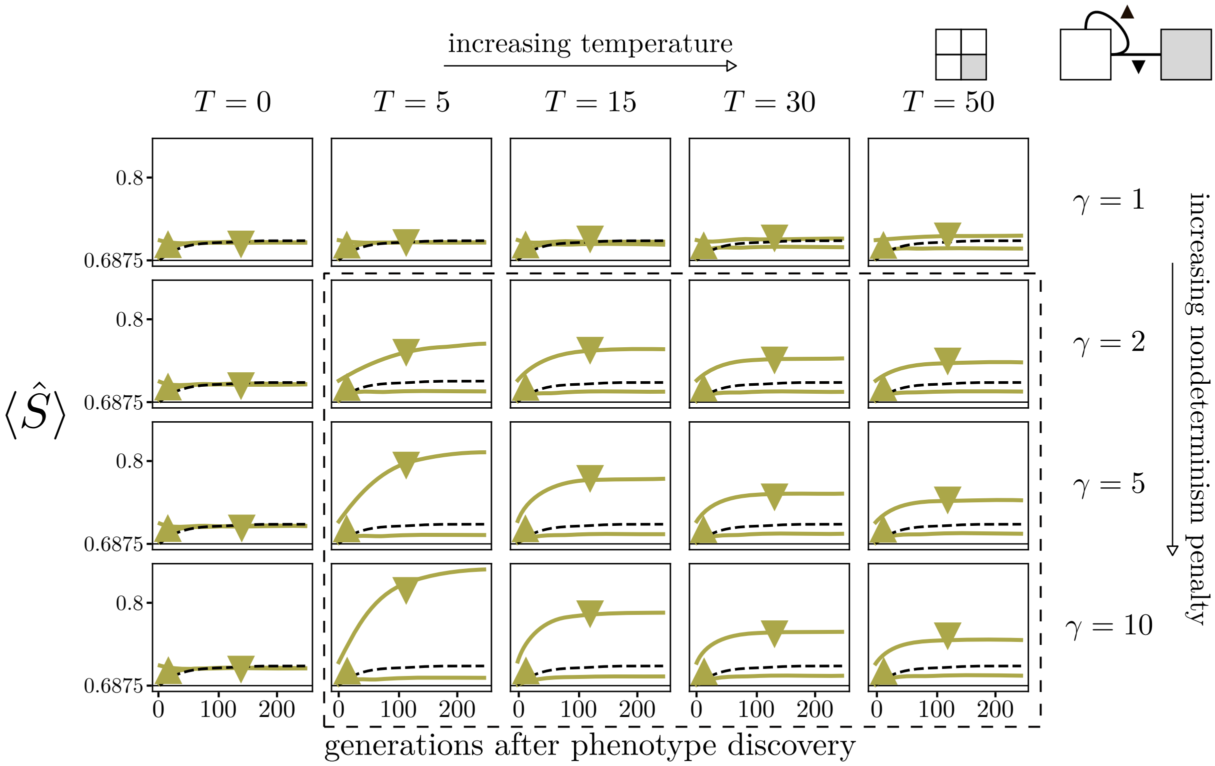


Figure 3.8: Existence of selection pressure originating from nondeterministic assemblies preferring a particular order of assembly is not tied to specific parameter values. Provided $T > 0$ and $\gamma > 1$ (black dashed box), where interaction strength impacts assembly order and nondeterminism loses fitness, the behaviour is universal. The magnitude of the gap depends on the selection pressure, which depends on parameters, but the existence only depends on interactions competing for assembly spots. This example is the heterotetramer evolving from the dimer (top middle panel of Figure 3.7), but similarly holds for any nondeterministic assembly.

There is a more nuanced version of this effect observed for $\gamma = 1$ at higher temperatures, where small fluctuations to the strengths cause the homotetramer to assembly most commonly, and hence change the phenotype. Since producing an incorrect phenotype causes a total loss of fitness, there is a second-order selection pressure to maintain correct assembly of the heterotetramer. The population evolves to maintain a strength gap just large enough to be robust to single mutations and promote the correct assembly ordering. Beyond that point, the second order selection pressure disappears and interactions are not driven further away from expectations.

3.4 Phenotype phase space

To thoroughly understand the evolution transition success rates and the nontrivial relationship of parameters observed in Figure 3.8, the competition between interactions must be explored further. In the case of the heterotetramer, the inter-subunit interaction is the only assembly step that can produce the correct phenotype, and so increasing the probability of that step increases the probability of correct assembly. However, this qualitative statement can easily be codified into an analytic relationship.

For more complex assemblies, such as the 12-mer with multiple competing interactions, this analytic relationship is central to understanding the more diverse dynamics involved in the transitions, and thus to the analysis of its evolutionary pathways. The mathematical description of the relative importance of the interactions is conceptually similar to a phase space. A given set of interaction strengths (the coordinates) most commonly produces one polyomino (the phase) while another set might produce something else entirely. These phase spaces can be challenging to map, but provide a comprehensive overview of the dynamics of nondeterministic assembly. Deriving the structure of these spaces can be highly challenging, but is justified by the information they contain for nondeterministic assemblies and associated dynamics.

3.4.1 Recursive assembly state enumeration

The probability of ending in a final assembly state, i.e. the resulting polyomino, is the sum of probabilities over all relevant assembly pathways. Each pathway's probability is in turn the product of probabilities for each assembly step. Assembly step probability is the interaction probability leading to that specific step normalised by the sum of all other relevant interaction probabilities. Assembly is carried out in this manner until termination, at which point all terms are collected to produce an analytic expression of the desired phase space.

Direct enumeration of steps is reasonable for smaller, simple phenotypes; unfortunately it quickly loses tractability. For a situation such as the 12-mer, where there are approximately 3 interactions possible at each step over 11 assembly steps, there are roughly $3^{11} \approx 10^{6}$ possible pathways to sum over. Fortunately, many of these pathways are similar and can be combined to reduce the number of terms involved.

Particularly for non-competing interactions, many assembly steps do not branch apart to unique final states, eventually coalescing to the same polyomino. As such, summing over these branches returns a unit probability for that step. Such a situation is also possible at different time steps, when branching at step T or $T+1$ doesn't lock off any final state, and the probabilities similarly sum to one. This realisation allows the exponential explosion of pathways to be reduced to a possible but probably laborious calculation.

Heterotetramer phase space

The phase space for the heterotetramer phenotype can be easily calculated through various techniques. After identifying symmetry invariant steps, where different interaction choices only correspond to an absolute rotation of the phenotype, only nine unique assembly steps remain. Directly enumerating these steps is perfectly reasonable. Even more simply, homotetramer formation only occurs when the intra-subunit interaction is picked three times in a row. Given interaction probabilities A and B for the intra- and inter-subunit interactions respectively, the probability of homotetramer formation is $P_{\text{hom}} = \left(\frac{2A}{2A+B}\right)^3/2$. Note this can only occur from one seed, hence the factor of 1/2. Since there are only two phenotypes possible, the heterotetramer probability is then the complement, $P_{\text{het}} = 1 - P_{\text{hom}}$, but this approach does not always generalise well.

Although the overall assembly pathway for the heterotetramer is nondeterministic, once the grey subunit from Figure 3.9 exists in the growing structure, pathways collapse to deterministically assembly the heterotetramer. Thus the first partial assembly state which contains the grey subunit, regardless of time step, can be thought of as a recurrent state. After entering this state, all possible assembly pathways will have the same known probability contribution to the final states. This approach is the most general framework for analysing more complex assembly trees due to the huge reduction in calculations after exploiting these similarities. An explicit demonstration of this approach and coalescing assembly steps is shown in Figure 3.9.

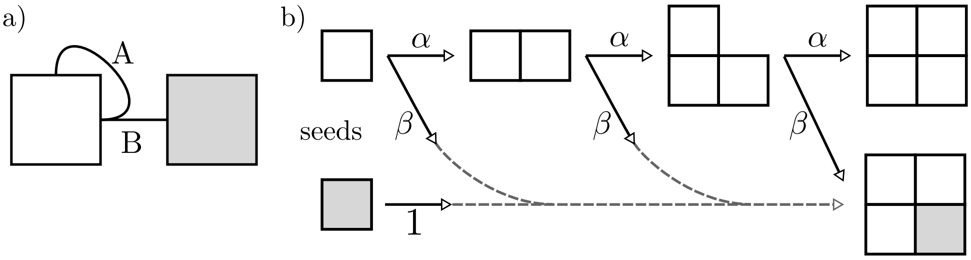


Figure 3.9: a) The assembly graph has interaction probabilities A and B for the intra- and inter-subunit interactions respectively. b) Once the grey subunit exists in the growing structure, assembly deterministically results in the heterotetramer. Intermediate steps can be skipped once such a recurrent state is identified, designated by the dashed lines. The normalised assembly tree probabilities are $\alpha = 2A/(2A+B)$ and $\beta = B/(2A+B)$.

Crucially, the actual interaction probabilities and thus strengths do not particularly matter. Phase space coordinates are naturally expressed in terms of normalised assembly tree probabilities, which are ratios of interaction probabilities and thus strengths. As these ratios change, the likelihood of producing a certain phenotype shifts as well. Linking interaction strength ratios and assembled phenotype provides *a priori* knowledge into evolution transition successes, explored in the subsequent section.

12-mer phase space

With more assembly steps and even more branchings at each level, the 12-mer phase space cannot be directly enumerated. There are also three possible polyominoes as final states: the homotetramer, 10-mer, and 12-mer. Calculating the simplest probability is thus insufficient to solve the entire system. Instead the recursive tree approach must be employed, making use of similar arguments as in the heterotetramer case to find recurrent states.

There are two main recurrent states for the 12-mer phase space, containing two and three tiles respectively. The second recurrent state is actually the first recurrent state plus an additional tile. These states are encountered when two interactions become non-competitive and all the potentially nondeterministic placed tiles become irrelevant. Assembly effectively "reverts" to an earlier stage, but that branch of the assembly tree keeps its probability factor. Both recurrent states and the complete tree leading to all three possible polyominoes are shown in Figure 3.10.

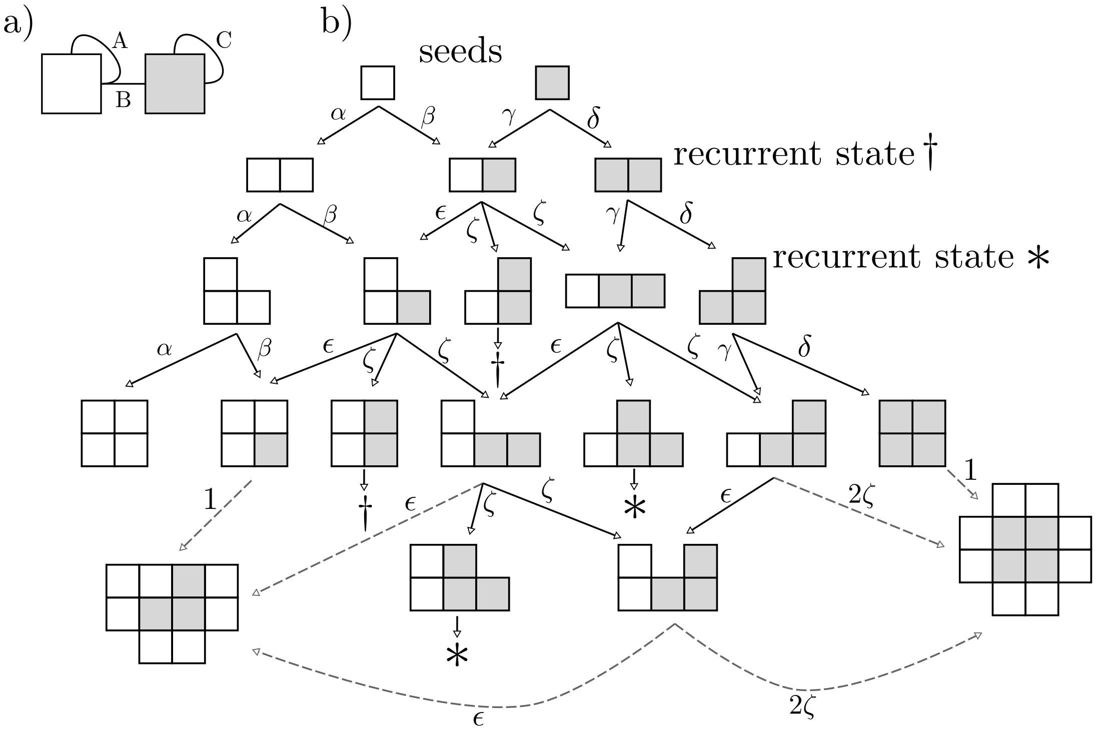


Figure 3.10: a) The assembly graph for the 12-mer has one interaction more than the example in Figure 3.9, but has a much more complex assembly tree. b) There are two recurrent state, identified with † and ∗. When these steps are reached, the assembly tree essentially jumps back to the recurrent state, but keeping the existing probability factors. Again the dashed lines again indicate lack of any competition, allowing intermediate steps to be fast-forwarded to the final polyomino. The normalised assembly tree probabilities are $\alpha = 2A/(2A+B)$, $\beta = B/(2A+B)$, $\gamma = B/(2C+B)$, $\delta = 2C/(2C+B)$, $\epsilon = A/(A+2C)$, and $\zeta = C/(2C+B)$.

Solving the homotetramer probability is straightforward, and is actually exactly the same calculation as the homotetramer from the heterotetramer example. This equivalence highlights the importance of analysing mesoscopic interaction competition rather than macroscopic phenotypes. On the other hand, even with the recursive tree, the calculations for the other two polyominoes involve over twenty-five terms in their most condensed form. Despite this complexity, only two interactions ever compete in a given step. This means the probabilities for forming any given polyomino can be expressed in terms of two ratios of the interaction probabilities, e.g. A/B and A/C or A/B and B/C etc.

While the heterotetramer phase space is quite easy to picture, with only two possible polyominoes, the 12-mer space is not. Since two ratios are sufficient to express the probabilities, and there are three polyominoes, the phase space can fortunately be represented in two-dimensional space using three channels of information, one per polyomino. A Red-Green-Blue (RGB) plot can encode the relative abundance of each polyomino for any given coordinate describing interaction strengths, shown in Figure 3.11. Appreciating the finer structure of the phase space unlocks insight into evolutionary dynamics on this landscape, as demonstrated later.

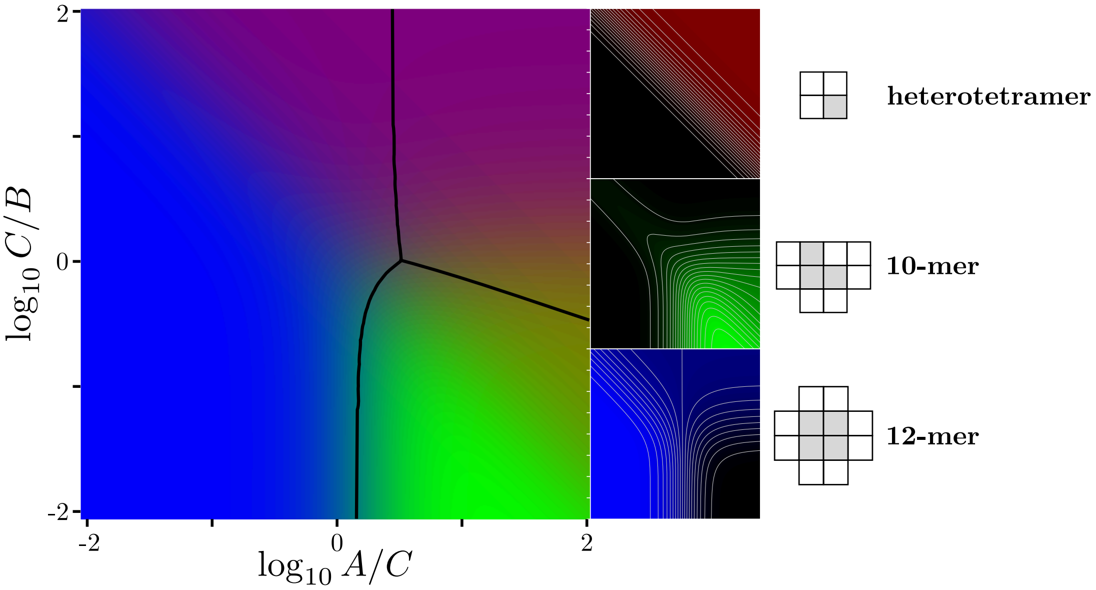


Figure 3.11: The 12-mer phase can be visualised with an RGB plot, with one colour channel per polyomino. Black lines indicate transitions where a different polyomino becomes the most common, and thus the phenotype. Individual channels are shown on the right for their respective polyomino, with frequency contours. Each polyomino's contours differ substantially, and so there is not an obvious outcome to altering interaction ratios in a particular way.

3.4.2 Phase space dimensionality

While the dimensionality of a standard phase space is determined by the degrees of freedom, it is not so simple in the phenotype case. Only interactions which compete in individual assembly steps add a dimension, and generally that dimension is a ratio of two strengths. Adding additional interactions, even competing interactions, would not necessarily increase the dimensionality.

Although such an assembly would be highly complex, it is plausible to imagine a situation where there are multiple sets of competing interactions that do not compete across the sets. In such a situation, the phase space could be two-dimensional despite having four competing interactions. Subsequently, there is no simple rule for determining dimensionality. Instead, the dimensionality must be determined by directly considering the minimum number of strength ratios required by the assembly tree. Provided any competing interaction ratio is somehow expressible, i.e. they span the phase space as a complete basis, the choice of ratios is arbitrary.

Additionally, there is no known general relationship between the number of interactions present in an assembly graph and the number of producible shapes. In strongly nondeterministic cases, where unbound growth is possible, just three interactions between two subunits are necessary to form an infinite number of polyominoes. While the number of producible polyominoes probably increases with more interactions in weakly nondeterministic assemblies, hence correlating dimensionality and phase count, it must be evaluated on a case-by-case basis.

This is precisely the situation for the 16-mer phase space, which is the largest polyomino assembled in this phenotype subset. Despite also having three interactions and even more assembly steps than the 12-mer, the phase space is in fact identical to that of the simple heterotetramer. As discussed earlier, one interaction in the 16-mer does not compete, and so does not contribute any dimensionality.

3.5 Evolutionary transitions

Individual transitions to more complex phenotypes are easy to identify, but defining the overall success rate is less so. Transitions in a population can be grouped into two main forms: fixations and failures. Fixating transitions are when a transition to a more complex phenotype survives, with descendants (who potentially evolve further) lasting the entire evolutionary history. Failing transitions, on the other hand, are when individuals of a more complex phenotype struggle to permeate the population and quickly go extinct despite a higher fitness potential. Some transitions fall between these two definitions, but are primarily artefacts of the finite population size and can be ignored explicitly.

Transition success rates, i.e. the fraction of fixating transitions, for the subset of six phenotypes are shown in Figure 3.12, calculated from the same evolutionary simulations as before. Successful transitions depend not only on the descendant's properties, but also on immediate ancestry. The transition to the heterotetramer from the dimer has a much higher success rate than the transition from the homotetramer, despite broad similarities. Both ancestors had one bond strengthened to the Markov expectation and one new minimally strong interaction, but their success rates are nearly a factor of three different.

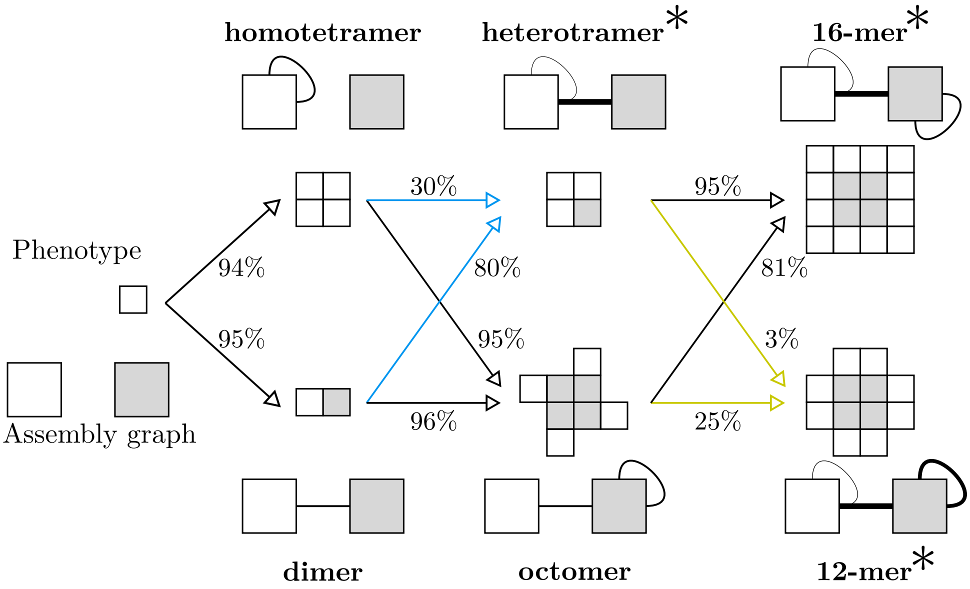


Figure 3.12: Transitions to deterministic assemblies have high success, with only a few failures due to stochastic effects of a finite population. Transitions to nondeterministic assemblies range from terrible to equally high, depending on interaction strength properties of their ancestors. Interactions widths highlight the optimal ordering of interaction strengths. Transitions to the heterotetramer (blue) and 12-mer (green) are explored in more detail in Figure 3.13.

3.5.1 Phase spaces and transition dynamics

This discrepancy can only be understood by considering the location of these transitions in the descendant's nondeterministic phase space. As seen earlier, competing interactions strengthen or weaken to maximise determinism according to selection pressures, which yields an equilibrium location in the phase space. Proximity to these coordinates determines the fate of newly transitioned phenotypes. If the ancestor's strength coordinates are too far from the optimal state of the descendant, the associated nondeterminism is more detrimental than the new phenotype is beneficial and overall the transition fails.

For all three nondeterministic cases, phase space transition locations suffice to explain every observed success rate. Explicit phase space transition locations are shown for the heterotetramer and 12-mer in Figure 3.13. In both cases, the ancestor which transitions nearer to the optimal descendant had better transition success.

Ancestors which are "pre-optimised" according to their interaction strengths at the moment of transition are those with the highest evolutionary success. Effectively, the nondeterminism is already partially minimised, and hence individuals get a greater portion of fitness which allows more successful transitions. This is most strongly observed in the case of transitioning from heterotetramer to 16-mer. The heterotetramer is already optimised by evolution to be as deterministic as allowed by the selection pressure. The heterotetramer and 16-mer have the same phase space, and so the transition occurs in the most optimal location and is quasi-deterministic. This explains the high transition success rate, which is in fact indistinguishable from truly deterministic pathways.

Alternatively, transitions from the octomer to the 16-mer are also mostly successful but to a lesser degree. After ignoring the interaction known to not compete, the transition

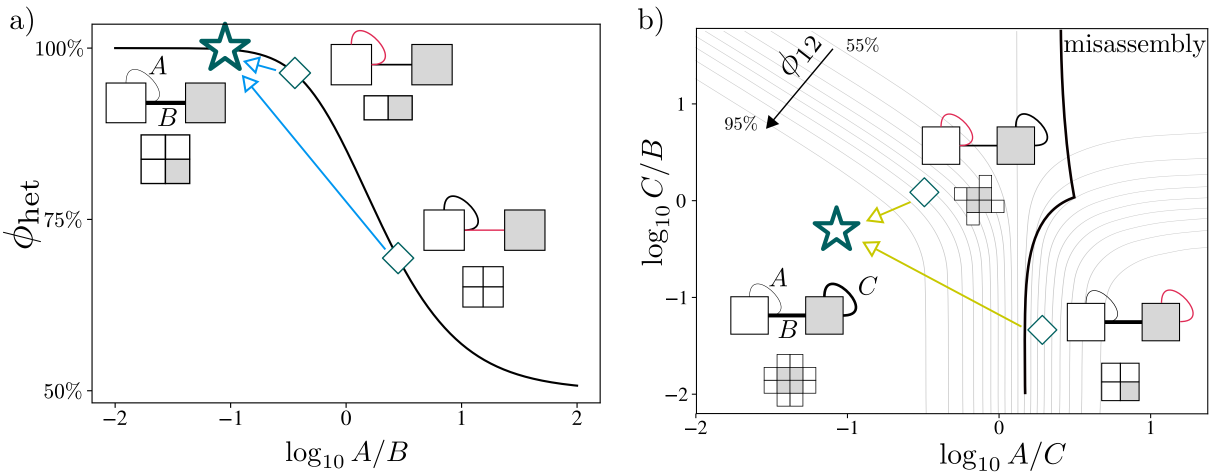


Figure 3.13: The phase location of transitions from ancestors are marked by diamonds. Their assembly graphs contain ancestral (black) and newly formed (red) interactions. The evolutionary optimum state for the descendent phenotype is marked by a star. a) The dimer is substantially higher up the determinism gradient ϕ_{het} than the homotetramer, and has correspondingly better transition success. b) The octomer transitions in a much better location to make the 12-mer compared to the heterotetramer. In fact, the heterotetramer most commonly assembles the 10-mer post-transition, as it is on the wrong side of the phenotype boundary.

of octomer to 16-mer is identical to the transition of dimer to heterotetramer. Both of these transitions have similar successes, at $\sim 80\%$.

Transitions to the 12-mer are once again more complex due to the higher dimensional phase space, but are still completely explainable. The octomer's interaction strengths are ordered more optimally compared to the heterotetramer, but still transitions down a steep determinism gradient with mediocre success. Conversely, the selection pressures operating on the heterotetramer drive the strengths in an sub-optimal direction, leading to frequently failed transitions. Generally the heterotetramer strength ratios commonly misassemble the 10-mer phenotype. It is only by fluctuations away from equilibrium, i.e. higher states in π_{PF}, can it produce the correct 12-mer phenotype, giving it a meagre 3% success rate.

Parameter invariance

Although there are many parameters involved in generating these results, the majority have simple qualitative effects. Changing any simulation parameter would likely impact the transition success rates in different ways. However, the numerical values of success rates are relatively unimportant, as the general dynamics behind them are known.

For example, changing the binary string length or strength threshold would change the steady state distribution π_{PF}. This distribution determines the fraction of the population fluctuating away from equilibrium, which may be needed so the fitness reward overcomes the nondeterminism resulting from the interaction probabilities. Alternatively, increasing the fitness reward would allow more individuals with greater nondeterminism to still have a net fitness advantage and successfully transition.

Transition success rates vary close to predictions for multiple parameter combinations, indicating any choice of parameter would produce results that are equally captured by the simple conceptual framework. Extended details on this can be found in Appendix C. This model can therefore be used to derive results about the evolution of nondeterministic self-assembly which are robust with respect to parameter choice. While not perfect, this assists translating insights from the model to how general forms of nondeterminism may have influenced evolution of interactions in biologically relevant contexts.

3.5.2 Predicting evolutionary pathways

The subset of phenotypes examined here was chosen due to the overall balance of abundances, interaction properties, and mutation accessibility. Even in general, armed only with combinatoric abundances and assuming arrival of the frequent [56], the likelihood of evolutionary pathways for any phenotype can be estimated. However, these two elements are often insufficient to explain the evolution of protein quaternary structure. For these simulations, likely pathways can be predicted *a priori* based on transitions where ancestors favourably reflect the assembly requirements of descendants. For example, the most likely pathway to the 16-mer is monomer → dimer → heterotetramer → 16-mer, while for the 12-mer is monomer → dimer/homotetramer → octomer → 12-mer.

The ordering of assembly steps in physical proteins can also be integral to producing the correct complex. This occurs on vastly different length scales, from misassembly induced from cotranslational protein folding [59] up to entire quaternary structures [48]. While some protein structures are guided by chaperones [49], others have evolved strong core interactions and weaker peripheral interactions [57].

In addition to strengthening interactions, gene fusion is another opportunity for proteins to cement a particular assembly order. Directly fusing subunits is similar to having an infinitely strong interaction which binds instantly, and hence can minimise the risk of misassembly in a similar manner [48].

Understanding evolutionary pathways and order of assembly is important to generally understanding complex formation, and is an active area of *in silico* [60] and bioinformatic [61] research. While evolutionary pathways and assembly pathways seem distinct concepts, there is strong evidence that assembly pathways are reflective of evolutionary pathways [47, 62], and so might be best considered concurrently.

3.5.3 Dynamic fitness landscapes

Much of the existing polyomino model literature and results have so far centred around evolving on a static fitness landscape. However, phenotype plasticity is a widespread phenomenon, where a genotype may be influenced by an environmental change resulting in some morphological adaptation. Nondeterministic assemblies can be interpreted as a form of plasticity, where multiple phenotypes can be produced from slight variations to the same interactions.

By incorporating a time-varying fitness landscape, alternatively rewarding fitness to different phenotypes, interaction strengths can continuously adapt. For the 12-mer example phenotype, one of its other assemblies, the 10-mer, is now also rewarded. Fitness is dynamically assigned as the sum of sinusoidal curves for these two polyominoes as

$$f_i = \sum_{p \in \mathcal{P}} \sin^2\left(\Omega t + \varphi_{\mathcal{P}}\right)$$

The two curves were set to be in antiphase, with $\varphi_{\text{10-mer}} = \pi$, and had a static oscillation period of $\Omega = 50$ generations. However, fitness could easily be assigned with a smaller phase gap or unequal periods if one phenotype was more important. As the fitness peaks change over time, selection pressure adjusts interaction strengths to follow, as seen in Figure 3.14. The population quickly settles into an optimal circuit, and is able to maximise dynamic fitness. Controllably modifying a phenotype in order to minimise nondeterminism is hugely advantageous to survival. The time scale for altering interaction strengths is linear in interface size, while forming new interactions requires quasi-exponential time.

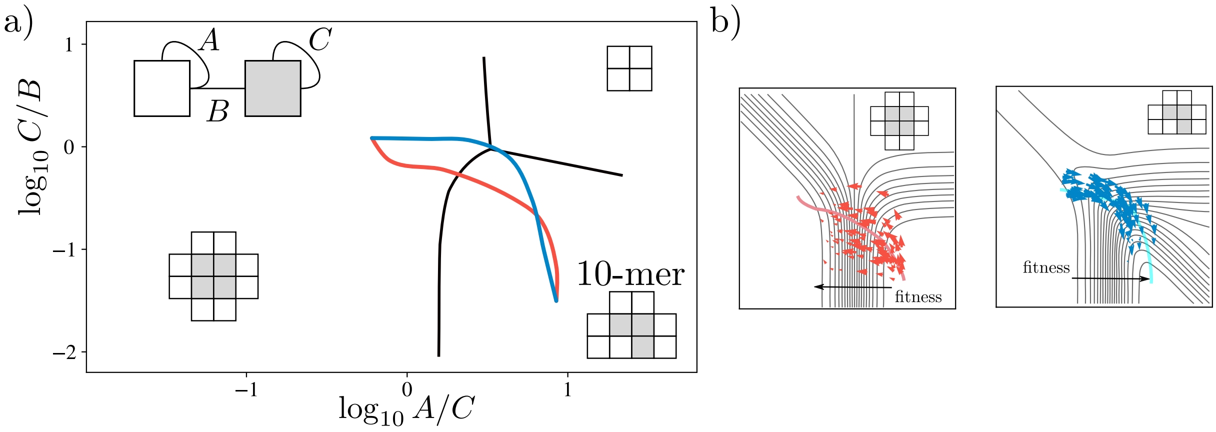


Figure 3.14: a) Slowly and smoothly alternating the fitness landscape to reward the 12-mer (red) and 10-mer (blue) allows evolution to optimally adjust interaction strengths. Any initial state eventually converges to the optimal closed cycle. b) Average trajectories follow the global determinism gradients closely, but some individual paths follow less efficient local gradients.

Populations generally followed the optimal determinism gradient. In the case of the 10-mer to 12-mer switch, the quickest route is to lower the A/C ratio, i.e. to move left in the phase space. Occasional interaction strength fluctuations lead some populations to an alternative route, first increasing the C/B ratio and then inverting the A/C ratio. This pathway took longer to achieve and thus had a lower average population fitness, but again highlights the nuanced role these phase spaces can play in understanding the evolution of interaction strengths.

The impact on evolution of plasticity through conformational change is uncertain, but emerging work suggests such behaviour imposes strong constraints on sequence evolution [63, 64]. Moreover, since assembly is commonly sensitive to adding or removing new interactions [11], merely modifying the strengths of existing interactions is likely to be a much safer avenue of adaption.

Interaction strengths themselves are also not the only opportunity to induce such changes. Stable complexes have been observed to form alternative structures when surrounded with an abundance of catalysts [65] or ligands [66]. The presence of these additional products modifies how the subunit interactions align, immediately resulting in different assembly. So plasticity enables much more rapid adaption to environmental changes, through conformational changes or reprioritising interactions.

Constant or immediate adaption in the polyomino model, as described above, barely scratches the surface of the interplay between plasticity and evolution. Multiple additions to the polyomino model would likely be required in order to explore plasticity generally, but it is conceivable to recreate these dynamics through a coarse-grained perspective. Such work remains an open avenue for further research discussed later in Chapter 5.

3.6 Conclusion

Polyomino self-assembly models have previously proven their worth in deepening the understanding of self-assembly phenomena and genotype-phenotype maps. Generalising the model's binding sites with binary strings retains tractability while extending applicability to other complex biological questions. Modelling the evolution of binding interaction strengths is one such example of qualitative insights beyond the reach of integer-labelled polyomino models.

In the absence of selection pressure, a Markov process approach provides an analytic expectation for the evolution of interaction strengths with minimal assumptions. These expectations agree well with simulations for deterministic phenotypes, but strongly diverge when time-ordering is important in nondeterministic assemblies. Such interactions are driven by selection pressure to strengthen or weaken, and thus assemble earlier or later, to maximise determinism. Certain evolution pathways are enhanced or inhibited based on relative strength ordering of interactions between ancestor and descendant, allowing a probabilistic reconstruction of most likely evolutionary histories.

Results taken from this model align with existing observations from experimental studies [46–48], as well as suggesting that nondeterminism as exemplified in the polyomino model could be a useful proxy for studying other manifestations of protein misassembly. Although the assembly kinematics are extremely simplified in the model, where stronger interactions tend to bind sooner, many proteins fall under this qualitative umbrella. However, protein complexes which rely on additional kinematic dynamics, such as chaperone-assited assembly [67], may have separate selection pressures resulting in different relationship between evolution order and assembly order.

Further study on topics briefly touched on, such as phenotype plasticity as well as other more complex genotype-phenotype mappings are imaginable within the generalised model.

CHAPTER

FOUR

DUPLICATION DRIVEN EVOLUTION PATHWAYS

Key messages

- Binary string self-interactions form much quicker than pairwise-interactions.
- Duplication-heteromerisation pathway enables self-interactions to transition to a more evolvable pairwise-interaction state.
- Examination of the Protein Data Bank validates several key qualitative model predictions.

This chapter is based on the manuscript in preparation under the working title *Duplication is a pathway for accelerating evolution of structural complexity* [68] by Leonard and Ahnert. All results are independent and novel work by Leonard under the supervision of Ahnert.

4.1 Introduction

Gene duplication, where a segment of existing genetic material is copied again, was first discussed in the 1970s, and has since been recognised as an indispensable process for biological evolution [69–71]. However, uncertainty still remains over the precise role that duplication plays, with many possibilities proposed over the years: creator of new genetic material [72], distributor of subfunctions [73], safeguard against extinction [74], or even just a passive passenger [70]. Whatever duplication's purpose, its prevalence is well-known on scales ranging from single genes to whole genomes.

Gene duplication appears more commonly in complex, eukaryotic organisms, whereas prokaryotes mainly evolve through other genetic mechanisms such as horizontal gene transfer [75]. Eukaryotes exhibit ample evidence of duplication driven evolution [71], with many ancestry-sharing homologous proteins, e.g. the kinetochore [76]. These homologous proteins can vary substantially in how robust they are against negative mutations [77] or evolvable towards positive ones [78], suggesting that duplication is not responsible for these properties by itself.

Symmetry is observed in the majority of multimeric protein complexes [79], and recent advances indicate that pseudosymmetric proteins may be more common than previously expected [80]. These pseudosymmetric protein complexes display strong homology between different subunits. Such a situation can readily occur under subfunctionalisation, specifically a form of duplication-degeneration-complementation [81].

4.2 Duplication driven pathways

Gene duplication does not directly introduce novel material; instead it merely copies existing genetic sequences. Any beneficial result of duplication should then come from subsequent evolution driven by neutral drift and selection pressures. Comparing properties and potentials of these evolution pathways when duplication is and is not allowed can tease out advantages that duplication may provide and help explain its prevalence throughout evolutionary history.

Duplicating a subunit which can interact with itself or other subunits increases the likelihood of that subunit binding during assembly. This shift in likelihood can distort the polyomino frequencies of nondeterministic assemblies, but this generally is a minor effect. However, mutating gene dosage in proteins can lead to deleterious stoichiometric imbalances [82], to the extent of requiring compensatory mechanisms in regulating some gene dosages [83].

Strikingly, subunit dosage naturally self-limits under the evolution simulation framework implemented here, without any direct fitness penalty. Excessive copies of a given subunit impart such a high robustness against negative mutations that they struggle to evolve through positive ones. Individuals with too few (zero) or too many (generally five or more) copies of a given subunit regularly are out-evolved by genotypes with an intermediate copy number. As such, genotypes with stronger dosage imbalances are selected against, while those with more balanced stoichiometry reproduce successfully.

Only genotypes with an intermediate number (more than zero, generally less than five) of duplicated subunit copies can then reliably evolve.

Before further considering the impact of duplication, it is crucial to understand the different nature of symmetric homomeric and heteromeric interactions, where a binding site interacts with itself or a different site respectively. There is a marked difference in interaction formation time, as well as in their effect on evolvability and robustness. Such differences drive the evolutionary advantage of duplication, as discussed later, with other associated dynamics only offering minor benefits.

4.2.1 Additional Markov chain interaction dynamics

The Markov chain formalism used for interaction strength evolution, detailed previously in Chapter 3, can be adapted much more broadly to model other interaction properties. Similar random walks approximate the time required for an interaction to form given random initial conditions or to be lost given a steady-state distribution. Much of the mathematical machinery remains unchanged, with the exception of the boundary condition at the critical strength. Interaction formation and loss is modelled by an *absorbing* Markov chain, as opposed to the *stationary* variant previously used. A schematic of this modified setup is shown in Figure 4.1.

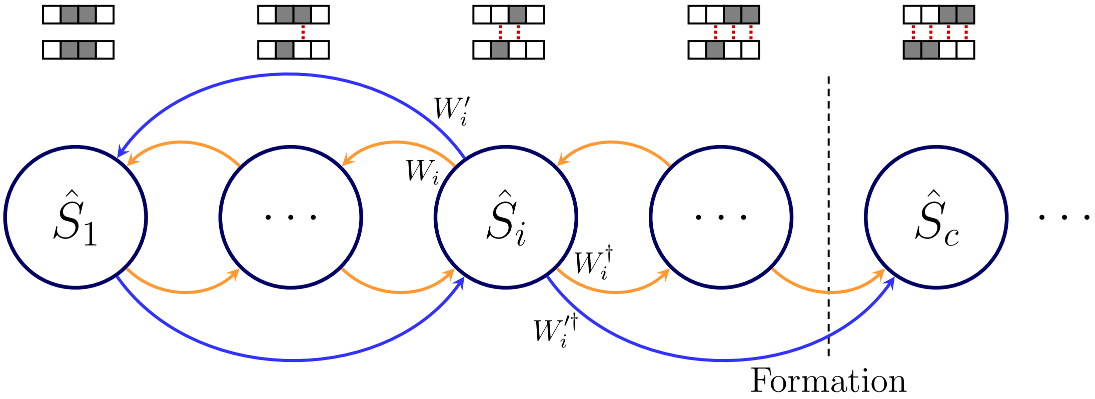


Figure 4.1: Interaction dynamics are still modelled by a Markov chain of strength states, with transition weights based on interaction strength. However, now interactions are formed or lost when they walk past the strength threshold and are "absorbed" (cf. Figure 3.4). Previously, only heteromeric (orange) interactions were considered. Symmetric homomeric (blue) interactions behave similarly, albeit larger steps.

Mutations to a symmetric homomeric interaction move by two strength states on the chain per step, as any point mutation affects the actual site and its partner counter-aligned site. Equivalently, symmetric homomeric interactions have an effective binding site length of $L' = L/2$, or half as many chain states. Otherwise their interaction strength dynamics are indistinguishable from heteromeric interactions.

Interaction formation expectation

Interactions are formed when two initial binary strings exist below the strength threshold, and then eventually mutate above it. Transition matrix weights are as previously defined, with W_i and $W_i^\dagger$ relating to the strength and complementary weakness of a given state. The critical strength state is still given by $c = \left\lceil L \cdot \hat{S}_c \right\rceil$, but now the matrix has $N = \left\lceil L \cdot \hat{S}_c \right\rceil + 1$ terms. Once an individual first drifts above the strength threshold and forms an interaction, selection dominates and there is no drifting back below the threshold, i.e. it is absorbed, indicated by the emphasised 0 transition weight.

$$\mathbf{F} = \underbrace{\begin{pmatrix} 0 & W_1 & & & \\ W_0^\dagger & 0 & \ddots & & \\ & \ddots & \ddots & W_{c-1} & \\ & & \ddots & \ddots & \mathbf{0} \\ & & & W_{c-1}^\dagger & 0 \end{pmatrix}}_{N}$$

The properties of absorbing Markov chains can be calculated using the fundamental matrix $\mathbf{N}$, which captures the probability of transitioning between states i and j after exactly k steps, summed over all k. Using the result for the infinite sum of a geometric series, the fundamental matrix can be expressed as $\mathbf{N} = (\mathbf{I} - \mathbf{F})^{-1}$, where $\mathbf{I}$ is an identity matrix. The number of expected steps to reach an absorbing state from any initial state is given by $\underline{t} = \mathbf{N}\underline{1}^\intercal$, where $\underline{1}$ is a vector of ones.

Reasonable choices of model parameters, i.e. sufficiently long binary strings and moderately high strength threshold, imply that any randomly initialised genotype will have strength states below the threshold. The probability to be in any given state is given by a binomial, and so the initial condition strength vector $\underline{B}$ is defined as $B_i = \binom{L}{i}(.5)^L$. The expected forming time is thus:

$$\langle \tau_{\text{form}} \rangle = \underline{B} \cdot \underline{t} = \underline{B} \cdot (\mathbf{I} - \mathbf{F})^{-1}\, \underline{1}^\intercal$$

Although the transition matrix appears simple, there is no general closed-form solution to the equation above. However, an approximate form was found through performing nonlinear regression on simulated interaction formation times. First, the L dependence was estimated over $L \in [50, 150]$. After the form of L was fixed, the $\hat{S}_c$ dependence was estimated over $\hat{S}_c \in [.6, .8]$. The resulting empirical form, providing a useful approximation over the relevant parameter window, is:

$$\langle \tau_{\text{form}} \rangle \approx \exp\left(\left(0.7L^{-1/2} + 0.08\right) \cdot L^{1.35\hat{S}_c}\right)$$

Regardless of the exact form, there is strong dependence on both L and $\hat{S}_c$ which appears quasi-exponential. Formation times quickly explode as these two parameters increase. Since symmetric homomeric interactions have an effective length of $L/2$, they routinely form significantly quicker than a heteromeric interaction with equivalent parameters.

Interaction loss expectation

Due to the simplicity of binary string binding sites, interaction loss can be modelled analogously to interaction formation. Completely reversing the Markov chain and modifying which state is the critical state produces a different transition matrix that is used identically to above. Now the absorbing state corresponds to when the interaction is lost and is assumed to never reform*.

Interaction strength Markov chains can be "reversed" with the anti-diagonal identity matrix†, $\mathbf{J}$, as $\mathbf{F}^{\text{rev}} = \mathbf{J}^{-1}\mathbf{F}\mathbf{J}$. Regardless of reversing the chain, the number of states now relates to how many strengths are above the threshold, unlike in formation. The matrix then takes the size of $N = \left\lfloor L \cdot \left(1 - \hat{S}_c\right)\right\rfloor + 1$, which is identical to the expression for N in Chapter 3.

Initial population states are not longer binomially distributed, but are driven by selection dynamics. The steady-state distribution is again given by $\underline{\pi}_{\text{PF}}$ from the previous Markov chain applies here, and can be assumed to be the initial distribution. The expected time to first drift below the strength threshold, and thus lose an interaction, is then:

$$\langle \tau_{\text{loss}} \rangle = \underline{\pi}_{\text{PF}} \cdot \underline{t}' = \underline{\pi}_{\text{PF}} \cdot \left(\mathbf{I} - \mathbf{F}^{\text{rev}}\right)^{-1} \underline{1}^{\intercal}$$

While the formalisms to calculate the expected formation and loss times share mathematical similarity, the resulting values differ greatly. For all relevant parameters used in this chapter, the expected time of formation is magnitudes greater than that of loss. Additionally, while heteromeric interactions are slower to form, they are also slower to be lost. This is despite the steady-state strength distribution for heteromeric interactions skewing closer to $\hat{S}_c$ ($\langle \hat{S} \rangle_L < \langle \hat{S} \rangle_{\text{L}/2}$).

4.2.2 Genotype complexity ceiling

Complexity can be difficult to define in biological contexts. Kolmogorov complexity [84] is defined as the length of the most concise algorithmic description of a given output. Self-assembling structures can naturally use such a definition, measuring the minimum genotype information necessary to build some structure. This has already been successfully applied to the polyomino model and protein complexes [19].

As outlined in Chapter 2, bound and deterministic assembly graphs can only support a limited number of interactions relative to the number of subunits, estimated by the first Betti number (see Section 2.2.4). This limit can be recast in terms of the maximum complexity an assembly graph can support, as more interactions require more information to encode. Beyond this complexity, misassembly dominates.

This is particularly applicable to assembly graphs with cycles, as multiple cycles readily misassemble. Since the most likely evolutionary pathway starts with forming a symmetric homomeric interaction—a rank 2 cycle—most genotypes then rapidly reach their complexity "ceiling". At this point, a large portion of possible additional interactions will induce deleterious behaviour. In order for a structure to evolve further, this ceiling has to be overcome.

[*] This is a comparably significant assumption, as the probability for the interaction to reform is approximately $\mu' \left(1 - \hat{S}_i\right)$ if the less-fit individual survives even one round of selection.

[†] Some conventions refer to this as the *exchange* matrix.

4.2.3 Duplication-heteromerisation process

Duplication can occur through multiple biological mechanisms, e.g. replication slippage, unequal crossing-over, etc. [85]. Certain mechanisms may be more likely to copy entire chromosomes or just short sequences, but ultimately can still only duplicate existing genotypic segments. In this model, duplication copies an entire subunit, similar to gene-level duplication, including all its interactions. Immediately after duplication there are then two identical copies of a subunit.

The original and duplicated subunits can then evolve and accumulate mutations independently. Importantly, duplicating a subunit containing a symmetric homomeric interaction results in at least three interactions: two duplicate symmetric homomeric and one heteromeric interaction between the two copies. Only one of these interactions is required to maintain the phenotype, allowing the other interactions to neutrally drift, potentially out of existence.

There are numerous pathways of different surviving interaction configurations and their strengths, but not all are equally robust or evolvable. As shown in Figure 4.2, pathways ending with only a heteromeric interaction can most advantageously evolve a new symmetric homomeric interaction, and so have the highest evolutionary potential. Oppositely, forming the additional symmetric homomeric interaction while the original symmetric homomeric interaction still exists can lead to unbound or nondeterministic assembly.

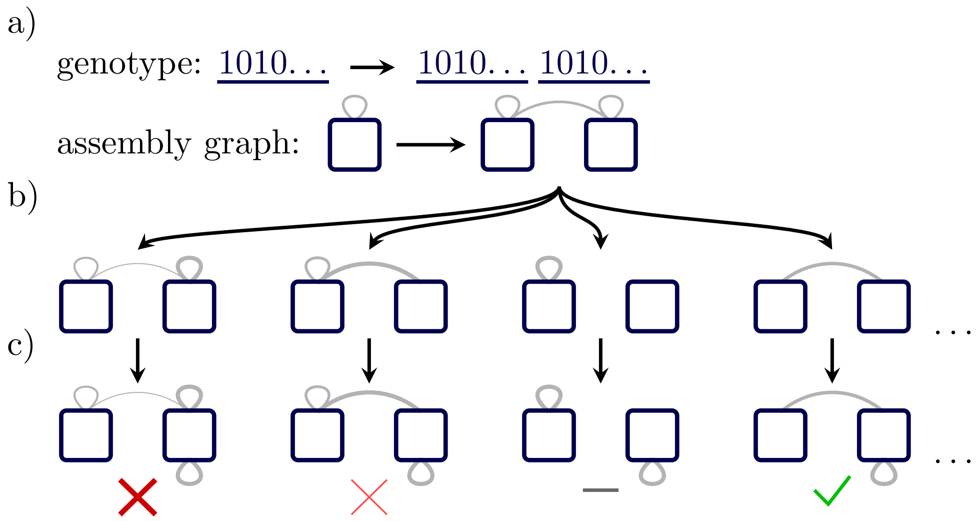


Figure 4.2: a) Duplication events replicate regions of the genotype encoding a single subunit, duplicating any interactions involving that subunit. b) There are many configurations in which mutations can accumulate on the existing interactions, with interaction width indicating strength. All possible drifts which preserve at least one interaction are neutral. c) Forming a new interaction breaks the neutrality. From left to right, the fitness change is highly negative (unbound & strongly nondeterministic), slightly negative (weakly nondeterministic), neutral (same phenotype), and positive (bound & deterministic, larger phenotype).

Duplication-heteromerisation allows a genotype to circumvent the complexity ceiling by transforming an assembly cycle into multiple single-use interactions. Such a genotype can now accept almost any new interaction, rather than being stuck at the bottleneck needing certain heteromeric interactions only. Few individuals successfully complete this process, as there is no direct selection pressure and many configurations of mutations ending in different states. Since the time scales involved in drift are minuscule compared to interaction formation, there are sufficient chances for a member in a population to evolve this way.

Crucially, this pathway is not limited to a single iteration, but can be repeatedly exploited by evolution. Duplication-heteromerisation on larger genotypes becomes more challenging as multiple subunits need to drift and subfunctionalise near simultaneously. Overall the pathway fades in advantage, but can still occur over longer time scales.

4.3 Evolution simulations

Evolutionary histories are simulated with a similar framework to that described in Chapter 3. A fixed-size haploid population reproduces asexually, with fitter individuals proportionally more likely to reproduce. Mutations again occur with a fixed probability per bit in the genotype. However, now duplication of an entire subunit can occur between reproduction and mutation with a fixed probability per subunit. In addition, any subunit can be deleted from the genotype with a probability comparable to the duplication rate, ensuring approximately constant genotype size. Details on these simulation parameters are provided in Table A.2.

Duplication and deletion introduce dynamic genotype length and potentially rapid turnover of subunits, and so tracking individual interactions over time is no longer meaningful. Instead, "degenerate" interactions should be viewed as an ensemble, e.g. a symmetric homomeric interaction and a duplicated copy are fundamentally related, and should be considered together in order to capture the entire dynamics acting on their evolution.

A simple approach to defining an ensemble is by the geometric position of the interactions on their subunits. For example, any interactions which occur between the top of a subunit and the top of a subunit are grouped together, or the left of one and the bottom of another. Any interactions which share geometric specifications are likely to be related through duplication of some shared ancestral interaction, which may have been lost to drift or deletion many generations ago.

This grouping strategy eliminates the vagueness of when duplicated interactions have diverged enough to be differentiable, or if an interaction drifts below and then back above the strength threshold which interrupts continuous tracking. Since simulations are constrained to evolving a small number of interactions, the chance that two independent interactions would evolve with the same geometric configuration is low and is an insignificant source of noise compared to the signal. Some examples of degenerate topologies are shown in Figure 4.3.

Interactions can be broadly grouped in three main groups: symmetric homomeric,

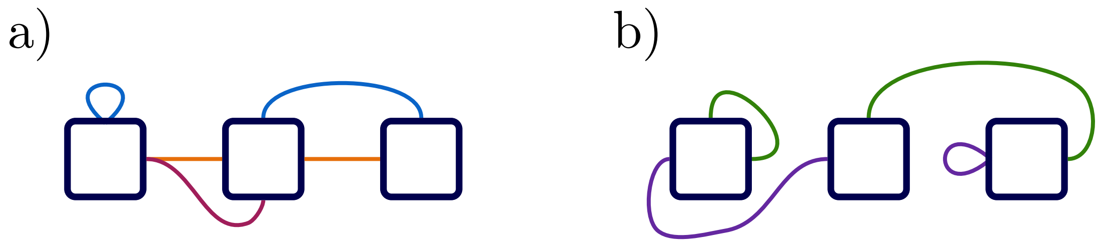


Figure 4.3: Two example assembly graphs with three and two interaction ensembles respectively in a) and b), where all interactions in an ensemble have the same colour. Two interactions can be degenerate even if one is intra-subunit and another is inter-subunit, due to the duplication-heteromerisation process. Any interaction properties, such as interaction strength or sequence similarity, should be considered over an entire degenerate ensemble, e.g. all purple-coloured interactions.

homologous heteromeric, and non-homologous heteromeric. Classifying between the two heteromeric interactions is based on ancestry, where homologous heteromeric interactions have shared history on the involved binding sites. Such history can only occur through subunit duplication. Non-homologous heteromeric interactions are the opposite, where there is no shared history between the binding sites and the interaction was formed *de novo*. Understanding the evolution and frequency of these three groups plays a crucial role in uncovering the purpose and impact of duplication.

4.3.1 Interaction homology and classification

Sequence similarity is a critical tool in identifying protein homology, measuring how many amino acids match or nearly match between two protein sequences. In the context of binary strings, interaction sequence similarity measures how identical two binding sites are through the Hamming distance, as the bits either match or are opposite. Sequence similarity shares conceptual overlap with interaction strength, as both related to the Hamming distance but with one string reversed in the latter case. As such, sequence similarity also can be predicted with a binomial distribution.

Two arbitrary binding sites rarely have strong sequence similarity, especially as length increases. Arbitrary sequence similarity actually directly follows a binomial distribution, $B\left(L,.5\right)$, yielding an average of $50 \pm \sqrt{1/4L}$ % sequence similarity. Symmetric homomeric interactions always have maximum sequence similarity, by definition, as they interact with themselves. Heteromeric interactions do not have such a constraint, and can take any possible value. However, finding a heteromeric interaction with high similarity ($\gg 50\%$) suggests they are not two arbitrary binding sites but instead have some shared evolutionary origin.

Tracking the ensemble distribution of interaction sequence similarities provides a simple marker of the duplication-heteromerisation pathway. Each interaction topology can be easily distinguished, allowing the evolution of similarities to be pinpointed for each ensemble. A specific topology ensemble which was initially observed with maximum similarity may later be observed fluctuating around $\sim 50\%$ similarity, indicating an initially symmetric homomeric interaction is now heteromeric. This transition is precisely predicted by the pathway, classifying the interaction as homologous heteromeric.

Alternatively, an ensemble may originate with medium similarity (indicating two arbitrary strings), classifying the interaction as non-homologous heteromeric. In this way, interactions can be assigned into the classes defined above. Two distinct example evolutions are shown in Figure 4.4, with all three possible interaction classes clearly identifiable.

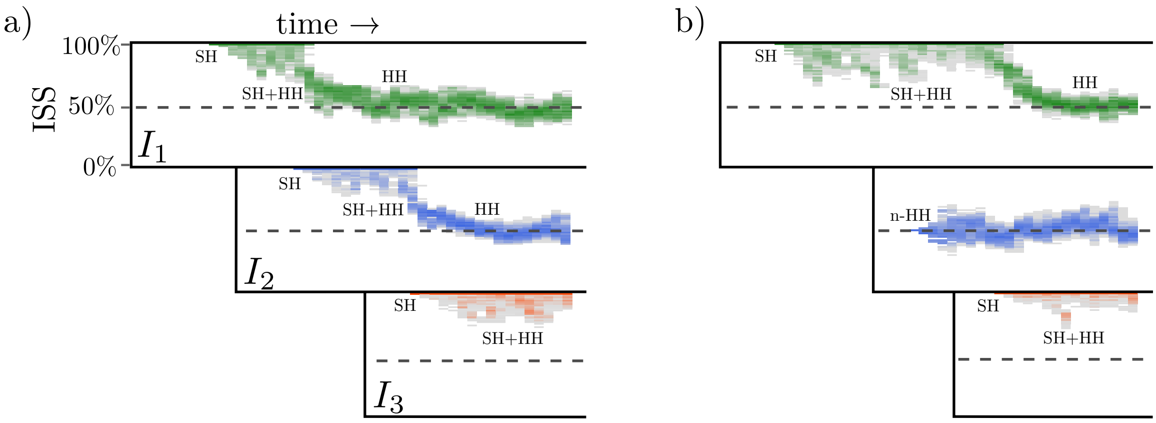


Figure 4.4: Interaction sequence similarity (ISS) can change over the course of evolution, shown here for three sequentially formed interactions (I_1, I_2, and I_3) over two independent simulations in a) and b). Each colour represents a unique interaction topology (see Figure 4.3). Lighter elements indicate that relatively few interactions of that topology across an entire population have that level of ISS at a given time, while darker elements indicate a higher population density in that state. Both simulations initially form a symmetric homomeric interaction (SH), which later coexists with its duplicate-drift homologous heteromeric interaction (HH). Forming a new symmetric homomeric interaction fixes the purely homologous heteromeric state. Note in b), forming a non-homologous heteromeric interaction (n-HH) as I_2 does not break the neutral symmetry of the SH+HH state, which continues to duplicate and drift until I_3 forms.

Duplication probabalistically occurs at all times, but is most noticeable after a symmetric homomeric interaction forms. Immediately after duplicating a subunit containing a symmetric homomeric interaction, heteromeric copies appear. As these heteromeric copies accumulate mutations and neutrally drift, their sequence similarity correspondingly drifts towards heteromeric expectations of 50%. These interactions can also be lost by drifting below the strength threshold, provided at least one interaction copy survives. This pattern of drift results in “smears” in sequence similarity, as heteromeric interaction differentiate themselves but then are lost to interaction or population dynamics.

Only when a new symmetric homomeric interaction forms is that neutral symmetry broken. At this stage, the heteromeric copy of an originally symmetric homomeric interaction quickly fixates in the population and fluctuates around the expected mean. The recently formed symmetric homomeric interaction then undergoes smearing drift, and the process can repeat. Note that forming an non-homologous heteromeric interaction (Figure 4.4b, I_2) has minimal impact on symmetric homomeric interactions which continue smearing drift.

4.3.2 Assembly graph interaction composition

Evolutionary simulations are highly stochastic, and the identification of single events is informative but insufficient to understand the diversity of dynamics at play. Averaging over multiple independent simulation provides a more comprehensive picture. For example, the composition of interaction types in the terminal evolved assembly graphs is relatively constant regardless of duplication; there is an approximately equal breakdown of symmetric homomeric and heteromeric interactions. However, the proportion of heteromeric interactions with shared evolutionary history is substantially larger in the event of duplication.

Interaction types, averaged over many assembly graphs, can easily be captured at every phenotype transition step, providing time-resolved tracking of composition. Without duplication, interactions are static and have no ability to change types, while with duplication only symmetric homomeric interactions can heteromerise. Although possible at any time, duplication only meaningfully influences subunits with existing interactions. Therefore, any consequences of duplication are only expected to arise after an interaction has formed.

Interaction compositions reflect this prediction, with symmetric homomeric interactions equally dominating the first interaction formation in both cases, seen in Figure 4.5. For later stages, however, they diverge significantly in behaviour. Most symmetric homomeric interactions undergo duplication-heteromerisation when possible, such that the second evolution step is again driven by symmetric homomeric formation and the original interactions become homologous heteromeric. Alternatively, without duplication, evolution can only proceed by forming non-homologous heteromeric interactions.

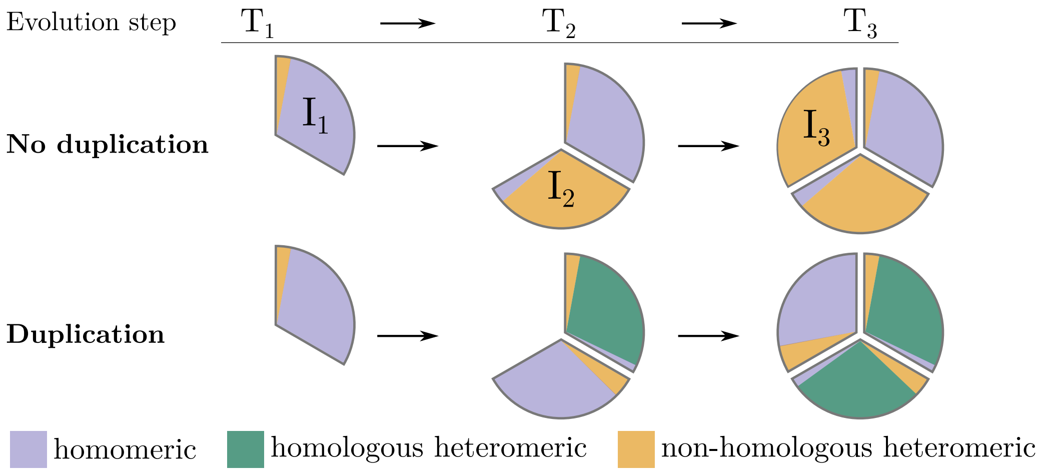


Figure 4.5: Assembly graph composition was recorded at each phenotype transition for the first three interaction (I_1, I_2, and I_3) formed at three sequential times (T_1, T_2, T_3 respectively), averaged over 1500 simulations. Without duplication, interaction types are static and do not change throughout evolution; later periods of evolution are predominantly driven by non-homologous heteromeric interactions. When duplication is allowed, symmetric homomeric interactions can transition to homologous heteromeric to facilitate evolvability, as strongly observed for the I_1 interaction at time T_2 and likewise I_2 at T_3.

After evolving through three phenotypes, the average assembly graph is comprised of about one symmetric homomeric and two heteromeric interactions regardless of duplication. As mentioned earlier, however, when duplication is allowed the heteromeric interactions are overwhelmingly homologous. The exact breakdowns depend on parameterisation, which shifts the relative advantage of the duplication-heteromerisation pathway over forming non-homologous heteromeric interactions. In general though, an abundance of homologous heteromeric interactions is a strong hallmark of duplication directed evolution.

4.3.3 Accelerating evolution of complexity

Tracking the number of interactions over time offers a reliable proxy to measuring the evolution rate of structural complexity, $\mathcal{C}$. Again due to the stochasticity inherent in single evolutions, averaging over many simulations provides a more stable trend that is suitable for interpretation. Although the Markov chain predictions for interaction formation have enormous variance, at times comparable to the mean, these expressions yield approximate checkpoints for the complexity quantity. The quicker symmetric homomeric interaction formation time, $\langle\tau^{L/2}_{\text{form}}\rangle$ should coincide with $\langle\mathcal{C}\rangle = 1$. The second interaction to form should be heteromeric, and so $\langle\mathcal{C}\rangle = 2$ is expected around $\langle\tau^{L}_{\text{form}}\rangle$.

As mentioned previously, duplication only realistically impacts evolution after the formation of the first interaction, and so the initial rates of evolving complexity should be comparable. Once duplication-heteromerisation comes into play, however, evolution should be accelerated with $\langle\mathcal{C}\rangle$ growing faster than expected. That means for a given period of evolution, simulations with duplication allowed should be at comparatively higher complexity. This is precisely observed in Figure 4.6.

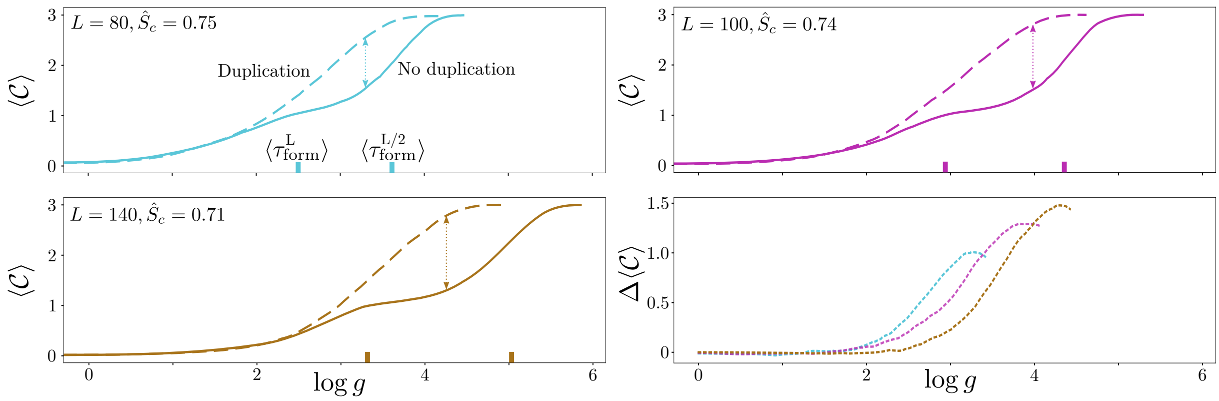


Figure 4.6: Structural complexity, $\mathcal{C}$, can be approximated by the number of interactions in an assembly graph. Trends are stable after averaging over 1500 simulations, up to a computationally imposed limit of three interactions. Note that a visually parallel line but at larger $\log g$ corresponds to a smaller slope due to logarithmically scaled generations. Duplication accelerates growth of complexity, measured by the gap between evolution ability with and without duplication, $\Delta\langle\mathcal{C}\rangle$. Evolution receives a greater boost as the duplication-heteromerisation becomes increasingly advantageous.

Evolutionary runs were restricted to a maximum of three interactions due to finite simulation resources, which is why the complexity asymptotically levels off at long time scales. As both sets of simulations converge to the imposed limit, the complexity gap diminishes to zero. However, smaller samples corroborate that the gap continuously widens when no constraint is placed, implying that duplication pathways repeatedly outpace standard evolution. Furthermore, comparable simulations using insertion of random subunits rather than duplication do not display any such acceleration. As such, the benefit appears localised to duplication-heteromerisation pathways rather than additional genetic length.

Importantly, as can be seen in the bottom right panel of Figure 4.6, the observed evolutionary advantage increases with binding site length L. This arises from the scaling "size" of the heteromeric bottleneck λ, where the time difference between forming a symmetric homomeric and heteromeric interaction diverges quasi-exponentially. Again there is no closed-form solution for formation times, but the ratio of the empirical form from Section 4.2.1 gives:

$$\lambda = \frac{\langle \tau_{\text{form}}^{L} \rangle}{\langle \tau_{\text{form}}^{L/2} \rangle} \approx \exp\left(0.2 L^{\left(1.35 \hat{S}_c - 0.5\right)}\right)$$

As L or $\hat{S}_c$ increase, the duplication pathway bypassing the bottleneck offers greater and greater gain. There is also dependence on duplication rate and effective population size, but these are approximately only linear effects.

With the nontrivial dependence on L and $\hat{S}_c$, different model parameters can have similar expectations for one interaction type but not the other. It is critical to analyse them separately, with the first formed interaction being overwhelmingly symmetric homomeric and the second heteromeric. Rescaling the required generations to form an interaction by these expectations provides a rough measure of "evolutionary epochs", which can be more directly compared across parameterisations.

Even after collating many simulations, the distribution of formation times has massive variance and makes histogram binning difficult. Since interaction formation is essentially a Bernoulli trial, the number of attempts before it forms can be approximated with a geometric distribution. As interaction formation typically takes many generations, this geometric distribution can be replaced with its continuous version: the exponential distribution. Fitting an exponential to the data agrees well with the noisy binned data, and so is a good approximate representation that simplifies interpretation. These exponentials are shown in Figure 4.7 for unscaled and rescaled generations.

As predicted earlier, duplication only accelerates evolution through the heteromerisation of symmetric homomeric interactions and no gain is observed in forming the first interaction. Once rescaled, all the different parameters approximately collapse onto a single line, substantiating the theoretical expectations and providing a general principle independent of parameter choice. Similarly for the second interaction, there is a collapse of simulations without duplication. However, simulations with duplication do not collapse at all, and take significantly less time to form, showing the evolutionary acceleration gained.

While the second interaction exponentials with duplication do not collapse, their

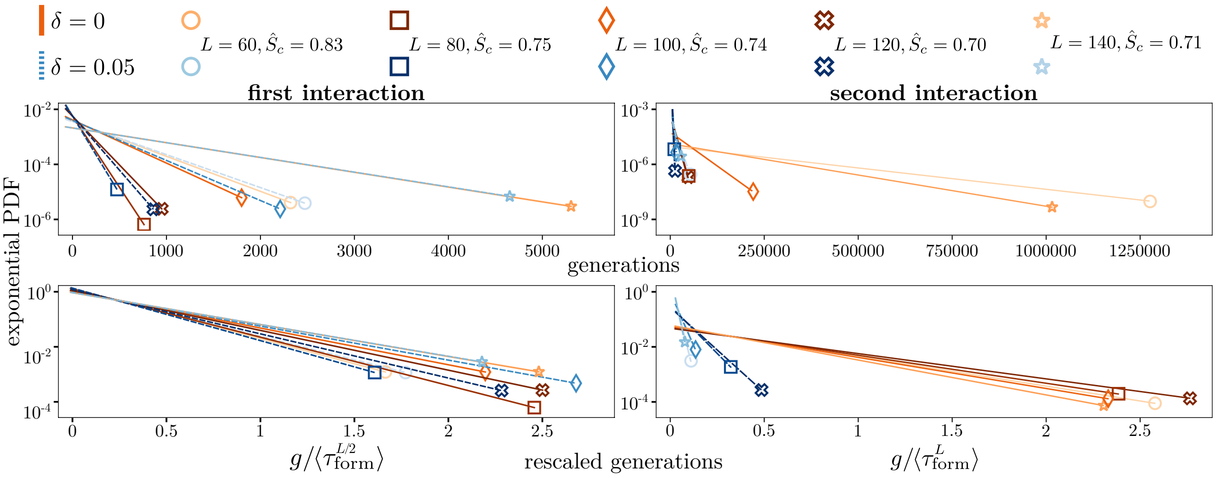


Figure 4.7: Fitted exponentials approximate the distribution of formation times well for various model parameters both without and with duplication, $\delta = 0$ and $\delta = 0.05$ respectively. Unscaled generations (top) can span many orders of magnitudes for the first (left) and second (right) interactions. After rescaling by the symmetric homomeric or heteromeric expectation, the majority of simulations collapse. However, the second interaction with duplication does not collapse, and is steeper than their without duplication counterparts—equivalent to faster formation.

ordering has analytical roots. Heteromeric bottleneck size λ, part of the advantage to duplication-heteromerisation, is the ratio of interaction formation expectation times as defined previously. As this ratio increases, duplication offers greater advantage. The second term involved in determining the evolutionary advantage is how frequently the results of the duplication-heteromerisation pathway survive genetic drift. This corresponds to when both symmetric homomeric interactions are lost but the heteromeric copy survives. Markov chains could be used again, with three correlated random walks: two of length $L/2$ and one of length L. An analytical result is difficult to derive, so these survival rates are most easily found empirically through separate simulations as ϵ.

The product of these two terms, $\lambda \cdot \epsilon$, is an estimate of the evolutionary advantage of the duplication-heteromerisation pathway. Values for these terms are provided in Table 4.1, showing how they depend nontrivially on choices of L and $\hat{S}_c$. The colour gradients in Figure 4.7 are determined by this product, with lighter colours corresponding to larger products and thus more advantage. Fitted exponential ordering reflects this, with lighter colours being steeper and thus forming the next interaction and evolving sooner.

Since the second interaction to form when duplication is allowed is typically symmetric homomeric, it is more appropriate to rescale these times by symmetric homomeric expectations. Expectations should be weighted by the survival rate ϵ, as not all duplication-heteromerisation attempts survive. The "fully" rescaled exponential fits are shown in Figure 4.8 for the second formed interactions.

Now all simulations collapse onto a single general line, regardless of binding site or interaction strength parameters or if duplication is allowed. While the theoretical expectations and empirical factors do not perfectly match simulations, the proposed pathway and underlying mechanisms reasonably explain all observations in the simulations.

Table 4.1: The bottleneck size λ varies significantly for different parameters of L and $\hat{S}_c$. Similarly, the heteromeric drift survival rate ϵ depends on both parameters, but it primarily increases with decreasing $\hat{S}_c$.

	Simulation parameters				
L	60	80	100	120	140
$\hat{S}_c$	0.83	0.75	0.74	0.70	0.71
λ-ratio	1923	246	537	184	1023
ϵ-factor	0.017	0.033	0.032	0.045	0.037
$\lambda \cdot \epsilon$	32.7	8.1	17.2	8.3	37.9

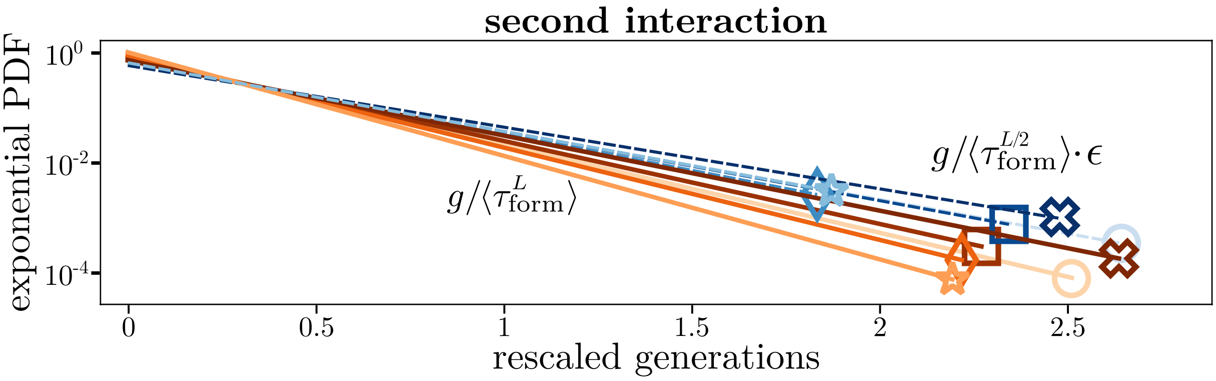


Figure 4.8: Rescaling the second interaction in the duplication simulations by the symmetric homomeric expectation and an empirical pathway survival factor ϵ collapses the distributions onto the previous collapse when for when duplication is not allowed. Colours and line properties are the same as from Figure 4.7.

As such, it is clear that the evolutionary advantage gained through duplication in these simulations originates from this specific proposed pathway.

4.4 Protein data bank

The lattice self-assembly model is a limited representation of real protein complexes. However, the simulations detailed above generate several qualitative hypotheses that can be compared to complexes taken from the Protein Data Bank (PDB) [86]. This data can be used to corroborate the hypotheses, even without access to the specific evolutionary histories of individual proteins.

Protein domains have been an important way to classify related proteins since their inception in 1973 by Watlaufer [87]. Domains are evolutionary conserved regions or compact units of folding that can form part of a larger protein subunit. Two distinct proteins sharing domains thus implies some evolutionary relationship between them.

CATH [88] is a major domain database which produces information on these evolution relationships. Protein subunits are assigned a designating CATH domain code for each domain present within them. Subunit can be assigned multiple different codes as well as multiple of the same code, depending on how many domains or copies of a domain the subunit has. If two unique protein subunits are assigned the same CATH domain

Table 4.2: There were 3658 heteromeric interactions present in the refined dataset. Homologous or non-homologous labels were assigned if there were or were not any overlapping CATH domains present on both interacting subunits. Subunit domain information was then randomly shuffled and labels reassigned, confidently showing homologous heteromeric interactions are over-represented.

		Domains	
		Homologous	Non-homologous
Pairing	CATH	1528	2130
	Shuffled	342	3316

code(s), they are in the same homologous superfamily and broadly overlap in terms of their evolutionary ancestry. Similar to the simulations, heteromeric interactions in protein complexes can be categorised as homologous or non-homologous if there is any overlap in domain information on the binding subunits.

The set of proteins analysed here is derived from a previously curated dataset [10]. All entries with at least one symmetric homomeric or heteromeric interaction are included from the original and extended sets of bijective protein complexes. Proteins that had missing or incomplete bioinformatic data, e.g. FASTA amino acid sequence, PDB atomic coordinates, CATH domains, etc., are removed. Redundant proteins interactions are filtered out using precompiled BLASTClust clusters (90% subunit sequence similarity, unless stated otherwise) available from the PDB. Only one representant interaction per combination of cluster IDs is allowed, picked randomly if multiple exist, to mitigate any abundance bias of proteins recorded in the PDB.

Similarly, protein complexes which contain many repetitious interactions, conceptually cycles in an assembly graph, have all interactions between identical subunits relabelled into a single interaction. If interaction sizes were slightly different between subunits, e.g. impacted by different conformations etc., an average value is taken. While there are many inherent biases and flaws that cannot be excluded within the databases, curated dataset, and data scrubbing methods, any overall trends present are reasonably robust enough to instil confidence in any conclusions.

4.4.1 Domain co-occurrence statistics

One of the simplest observations is the excessive correlation of domains across heteromeric interactions, i.e. the overabundance of homologous heteromers. A permutation test, randomly shuffling the domain assignments on each subunit and recalculating the quantity of homologous and non-homologous heteromeric interactions, is shown in Table 4.2. Fisher's exact test yields an overwhelmingly significant confidence, $p \sim 5 \times 10^{-235}$, that subunits involved in heteromeric interactions are more commonly homologous than naively expected. Additionally, only about 4% of domains in homologous heteromeric interactions did not also exist in a symmetric homomeric interaction, while it was double that for the non-homologous case.

In addition, many protein complexes are comprised of a mixture of interaction types. Approximately one quarter of complexes containing at least one heteromeric interaction also had a least one symmetric homomeric interaction. Out of 594 complexes with multiple heteromeric interactions, 73% of them contained a homologous heteromeric interaction. The average composition for these 594 proteins was about 2.0 homologous and 1.5 non-homologous heteromeric interaction, with $p = 6.6 \times 10^{-12}$ that any complex has a greater likelihood to have more homologous than non-homologous interactions. Such results qualitatively agree with the model simulations, showing that homologous heteromers are a significant contributor to the evolution of larger complexes.

Several assumptions in the model do not realistically reflect physical proteins. Specifically within the model, the number of possible heteromeric interactions scales quadratically with genetic length, compared to linearly for symmetric homomeric interactions. This relationship is combinatorially true for proteins as well. However, these two quantities are comparable when model genotypes only contain a handful of subunits, whereas proteins generally have magnitudes more possible pairwise combinations. So while the duplication-heteromerisation pathway can be orders of magnitude faster, non-homologous heteromeric interactions can still readily form given the quadratically scaling number of protein-protein interactions possible in close proximity, e.g. outside a cell nucleus etc. Non-homologous interactions outnumbering homologous in the data therefore is not in disagreement with the simulations, but reflective of a more varied environment.

4.4.2 Interaction buried surface area

One clear trend from model simulations is that longer binding site lengths L and higher strength threshold $\hat{S}_c$ result in a higher proportion of homologous heteromeric interactions. While the physical binding of protein subunits is likely based more on kinematics and interaction energies of their buried surface area (BSA), the two model parameters L and $\hat{S}_c$ controlling interactions are reasonable as a first-order approximation. BSA is influenced by many additional factors, such as steric exclusion or cooperative binding, but correlates well with binding affinity [89].

Without the ability to rerun evolution, the protein data cannot show that larger protein binding sites more frequently evolved via homologous heteromeric interactions. However, assessing the homology and BSA of heteromeric interactions taken from the protein data allows the converse question to be asked: are larger BSA heteromeric interactions more commonly homologous or non-homologous? Although the arguments seem similar, this question does not isolate the influence of BSA size to interaction binding, and results should be interpreted cautiously. Nonetheless, the gathered results clearly display a statistically larger BSA (“$\propto L$”) for homologous compared to non-homologous heteromers seen in Figure 4.9, supporting a key prediction from the model.

The data contains many homologous heteromeric interactions related to immunoglobulin proteins with high BSA, present in the upper peak. This immunoglobulin domain (CATH:

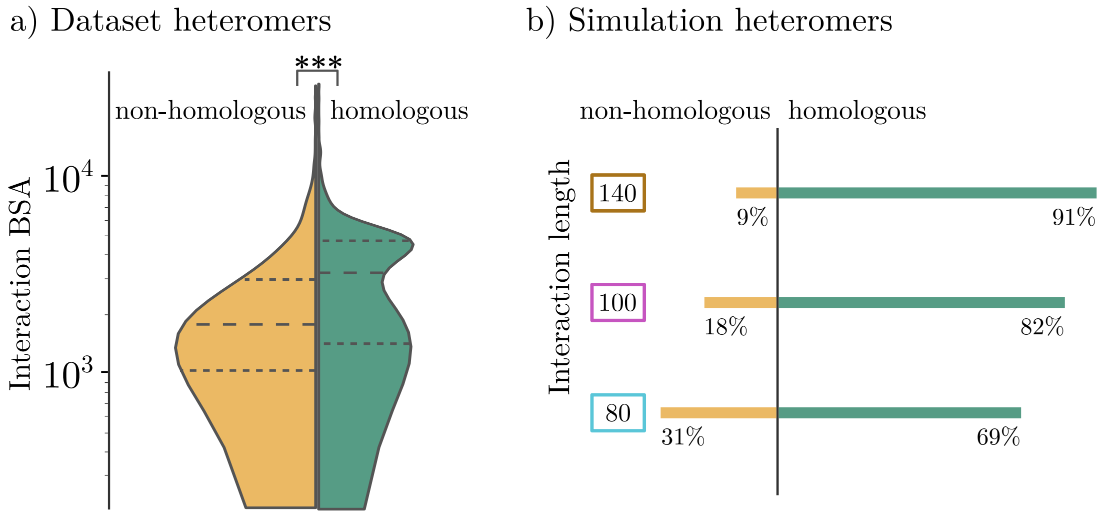


Figure 4.9: a) Homologous heteromeric protein interactions have statistically larger buried surface area than non-homologous interactions according to a Brunner-Munzel test: $p \sim 10^{-47}$. The common language effect size (CLES) is .64, i.e. the homologous BSA will be larger in 64 out of 100 random pairs. b) Heteromeric interactions from *in silico* simulations skew towards homologous ancestry as L increases, similar to the dataset results. Non-homologous heteromeric interactions can still form at long interaction lengths, but quickly become rare.

2.60.40.10) is the most common recorded domain, both in the dataset as well as in CATH. Statistical significance is still retained after removing any protein subunits with immunoglobulin domains ($p \approx 7 \times 10^{-10}$, CLES=.59), filtering at a 70% cluster similarity threshold ($p \approx 5 \times 10^{-9}$, CLES=.57), as well as combinations of these filters.

Not all factors can be accounted for, nor is BSA a direct analogue to binding site length. Regardless, the statistical significance is strong supporting evidence for the abundance of heteromeric interactions which likely evolved through duplication. Care should also be taken that the evidence is based on correlation rather than causation. However, these results are sufficient to justify a more direct method of testing for duplication-heteromerisation driven evolution.

4.4.3 Subunit binding sequence alignment

Alongside the structural and atomic coordinate data from the PDB and interaction homology from CATH, residue-level details from a database of crystallographic interactions (PDBePISA [90]) are critical to substantiating the duplication-heteromerisation pathway. Examining the specific residues involved in heteromeric interactions, where two subunits may have an excessive similarity, yields more incontrovertible proof that the interaction was once symmetric homomeric and has since duplicated and diverged.

There are two separate components in the interaction comparison: sequence alignment and binding residues. Sequence alignment is straightforward, using the Needleman–Wunsch algorithm as implemented by EMBOSS needle [91] with default parameters. Binding residues are taken from the macromolecular interfaces from PDBePISA and converted to an intermediate format to track binding residues by subunit. The two datastreams are then overlaid, such that binding residues are aligned according to optimal sequence matching. A diagrammatic overview of this **Su**bunit **B**inding **Se**quence **A**lignment (SuBSeA) process is shown in Figure 4.10.

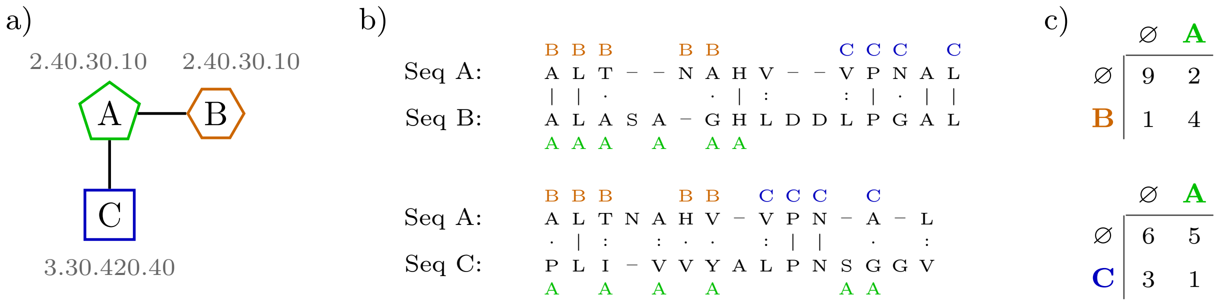


Figure 4.10: a) A trimeric protein where subunits A and B have a homologous heteromeric interaction while A and C have a non-homologous one, according to their example CATH domains in grey. b) After sequences are aligned, residues are clearly marked if they are involved in an interaction with another subunit. Identical, similar, and unrelated residues are indicated by '|', ':', and '.' respectively. c) Aligned binding residue partners are counted, e.g. how many times a 'C'-interacting residue is opposite an 'A'-interacting residue, or a 'B'-interacting residue opposite a non-interacting residue (denoted '∅'). The results are stored in the matrix **M**.

Co-occurrences of binding residues are then tracked in a matrix **M**, where each element is the number of aligned residues between two given subunits. In the common situation where one residue is not aligned opposite another target-interacting residue, it is tracked in a catch-all null column/row, ∅. The co-occurrence counts and the marginal distributions approximate the probability that the alignment of an interaction could happen by chance. The probability for an element in the matrix is given by:

$$p_{ij} = F\left(M_{ij}; \delta L, \frac{\sum_j M_{ij} \cdot \sum_i M_{ij}}{L_1 \cdot L_2}\right)$$

where F is the cumulative binomial distribution, L_1 and L_2 are the sequence lengths of the first and second subunits, and δL is the length of the overlapping alignment region.

If only null column and rows are present, the interaction is trivially assigned a unit p-value. Otherwise, the lowest probability (inferring the strongest likelihood of two interfaces being related) is stored and the relevant row and column are removed from the matrix. This process continues iteratively until there are no surviving rows or columns to sustain the matrix. The series of stored p-values is then combined using the Fisher method, although other methods can be used, e.g. Stouffer's, to determine the overall probability of interaction homology between two heteromerically linked subunits. Although the two interactions demonstrated comparable sequence similarity in Figure 4.10, the A-B interaction has more specific matching of residues compared to A-C, giving it higher SuBSeA confidence and validating the homologous label.

Confidences are assigned for each heteromeric interaction, which are grouped into homologous or non-homologous if they do or do not share domains, as previously. Survival functions for the distributions of these two groups are shown in Figure 4.11. These functions measures how much of a population is above some confidence level, and are related to cumulative distribution functions as SF = 1 − CDF. Crucially, homologous heteromeric interactions display substantially stronger confidences, giving credence to ancestral symmetric homomeric interactions which explicitly underwent duplication-heteromerisation.

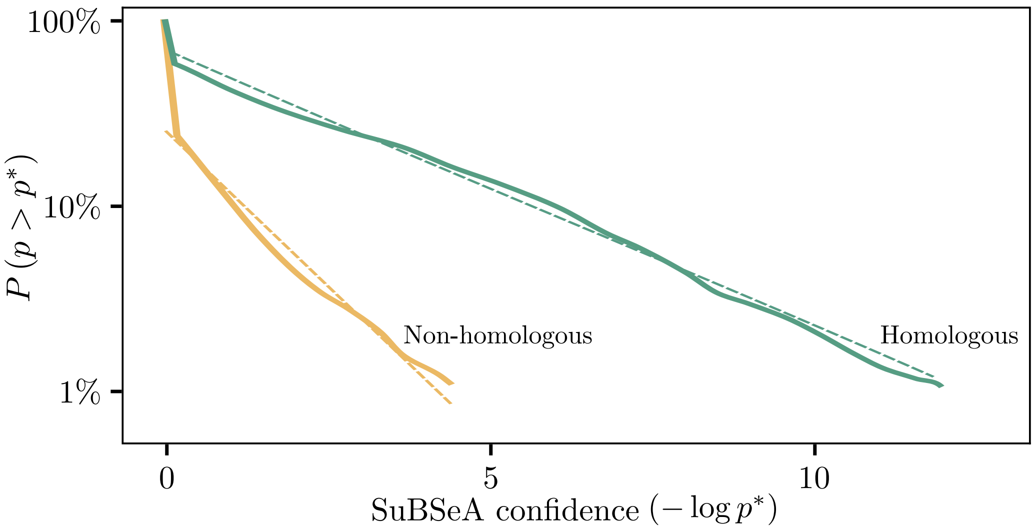


Figure 4.11: SuBSeA confidences calculated for 1528 homologous and 2130 non-homologous heteromeric interactions from the dataset. Homologous interactions displayed stronger confidences, evidenced by the slower decaying survival function, i.e. more homologous interactions had confidences greater than any arbitrary p^* compared to non-homologous interactions. The survival functions can be approximated with a power law for $-\log p^* > 0$ using the form $\propto p^{*-\xi}$, with exponents of 0.15 and 0.34 respectively.

Fewer homologous heteromeric interactions failed to find any alignment, which results in $\log p^* = 0$, compared to non-homologous interactions, 42% and 75% respectively. In addition, the power law approximation shows the SuBSeA confidence in homologous heteromeric interactions decays roughly twice as slowly as the non-homologous case. Nearly a third of the homologous alignments were more significant than the common threshold of $p^* = 0.05$ $(-\log p^* \approx 1.3)$, compared to only about 10% for non-homologous interactions.

Several interactions in the non-homologous distribution tail highlight the uncertainty in any classification scheme. For example, the largest outlier, PDBid: 3F3P, has single digit sequence similarity between subunits J and L but substantial structural similarity around the interacting region and a SuBSeA confidence of 10^{-23}. These subunits may have duplicated and diverged too far in the past to be assigned homologous domains, with subunit J and L having "propeller" and "sheet" domain architectures respectively. While domain assignments always on rely on incomplete knowledge, these observations suggest that the derivation of evolutionary ancestry can benefit from the combination of multiple sources of information [92].

4.4.4 Symmetric homomeric precursors

Additional confirmation of duplication-heteromerisation comes from comparing symmetric homomeric and heteromeric interactions. This indirect approach, which is more difficult to configure consistently, is based on whether heteromeric interactions have symmetric homomeric precursors. If a heteromeric interaction did evolve from a duplicated symmetric homomeric interaction, the symmetric homomeric version could still exist in another recorded protein complex. Unfortunately, it is also possible the symmetric homomeric version is nonextant or has not yet been recorded in the PDB.

Similarly, it is ambiguous when an interaction between two subunits should be classified as symmetric homomeric or heteromeric. An interaction with high sequence similarity could indicate "recently" duplicated subunits or two identical subunits with slight measuring/recording errors. Distinguishing between these two cases is not straightforward due to inherent experimental uncertainty and lack of a unified data standard, but should be manageable background noise to a stronger signal.

Comparisons are generated using the heteromeric interactions classified as homologous or non-homologous. Both subunits in each heteromeric interaction are compared against all symmetric homomeric subunits in the dataset which share domains. This produces an order of magnitude more comparisons than the direct heteromeric alignments. Overall, homologous heteromeric interactions still exhibit better alignment against symmetric homomeric complexes, shown in Figure 4.12.

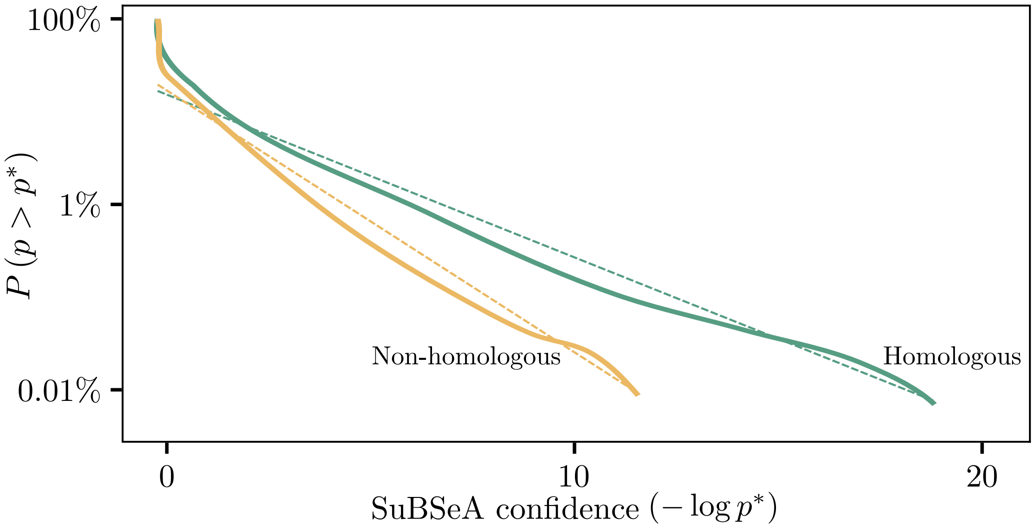


Figure 4.12: Subunits with heteromeric interactions can be compared against symmetric homomeric complexes which share CATH domains. As in Figure 4.11, binding alignments which are homologous are statistically more confident. Since there are many more symmetric homomeric-heteromeric comparisons possible, the survival function is stable down to a lower percentage and higher p^*. The ξ fits (see Figure 4.11) are approximately 0.22 and 0.33 respectively.

Despite the differences in the heteromeric and symmetric homomeric precursor and the direct heteromeric comparison, the results of the binding alignment are quite similar. The percentages of interactions failing to find any alignment were nearly identical to before, as 46% and 73% for homologous and non-homologous respectively (cf. 42% and 75%).

The power law approximations were also similar with 0.22 and 0.33 (cf. 0.15 and 0.34), although the homologous case has a more pronounced difference.

The non-homologous comparisons are almost the same, perhaps arising from some common short residue sequences that occur in many proteins or as a limit of how correlated sequences might be. Reassuringly, the homologous alignments are definitively better in the direct heteromeric comparison case. So while the precursor comparison provides an additional contribution to the likelihood of duplication-heteromerisation, the indirect case does indeed accumulate more noise in the alignments.

4.4.5 Duplicating arbitrary protein complex symmetry

Symmetric homomeric interactions are an incredibly widespread type of interaction [93], and have been the sole focus of the duplication-heteromerisation pathway thus far. These interactions impart the protein complex with C_2 symmetry; however, the concepts outlined here apply far more generally. With the direct relationship between assembly graph cycles and potential symmetry, reducing or eliminating an assembly graph cycle effectively reduces a complex's symmetry. Hence heteromerising the symmetric homomeric cycle ($C_2 \to C_1$ symmetry) improves evolvability, since cycles are a dominant contributor to the risk of unbound growth and strong nondeterminism [24].

Duplication-heteromerisation can act generally on any cyclic assembly to reduce the inherent symmetry through subfunctionalisation of the involved interactions. A homotetramer complex (C_4) can transition to a homodimer of heterodimers (loosely $C_2 \times C_2^{\text{pseudo}}$), where the heterodimer is comprised of homologous subunits. This complex can transition even further to an asymmetric complex of four homologous heteromeric subunits.

Such behaviour can conceptually be extended to three dimensions and other symmetries, e.g. dihedral groups. As the duplication-heteromerisation pathway relies on neutral drift, these more complex cycles and symmetries almost certainly reach the evolvable heteromeric state less frequently. However, this is just a rescaling factor of the time spent in the drifting phase rather than any mechanistic difference.

Thousands of likely candidates are easily identified through browsing the PDB by pseudosymmetries. One example case is the complex PDBid: 3WJM, with two unique subunits forming a heterohexamer with D_3 pseudosymmetry. Both unique subunits share the same two CATH domains and have just below 80% sequence similarity. These subunits have SuBSeA confidences between the different subunit interaction conformations of 4×10^{-7} to 2×10^{-34}. As such, this "dimer of trimers" probably evolved through cyclic interactions which have since duplicated and diverged into heteromeric states. This "X of Y" composition of complexes has been widely observed [94, 95], and the proposed pathway is entirely complementary to these existing studies.

Another example is the complex PDBid: 2IX2, which forms a heterotrimer from three unique subunits but with C_3 pseudosymmetry. The three subunits average around 50%

sequence similarity between them. This does imply homology, typically taken above 30% similarity, but is relatively weak evidence [96]. Comparing the interactions directly gives SuBSeA confidences of around 10^{-7}, providing stronger evidence of shared ancestry. Furthermore, this is an example of an asymmetric homomeric interaction and a symmetry that is not possible in square geometry. Both of these examples are shown in Figure 4.13.

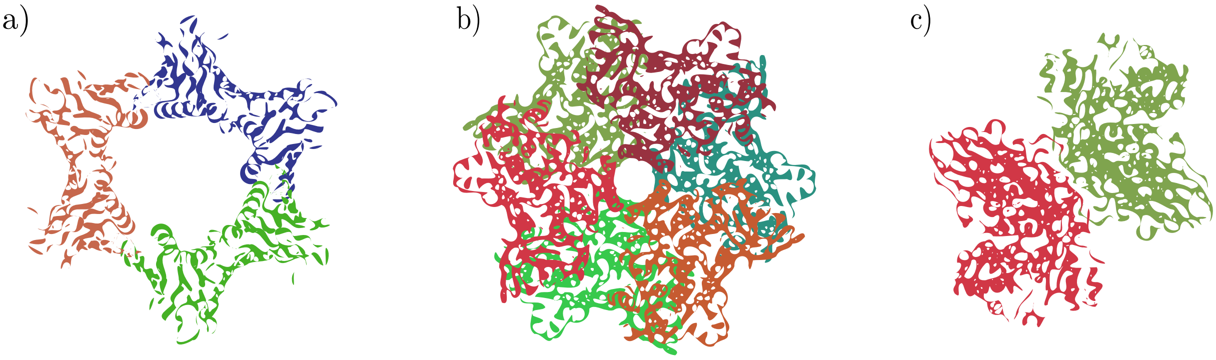


Figure 4.13: Two example protein complexes found through the PDB's pseudosymmetry browser. a) PDBid: 2IX2 has C_3 symmetry, with heteromeric interactions between three distinct subunits. However, sequence similarity and SuBSeA confidences suggest this was originally an asymmetric homomeric interaction. b) PDBid: 3WJM has D_3 symmetry, where two C_3 homotrimers form a dimeric complex. c) Side view of PDBid: 3WJM with only one subunit from each homotrimer. The heteromeric interaction forming the dimer again has strong SuBSeA confidences, implying it was once symmetric homomeric.

4.5 Conclusion

Genetic duplication is involved in evolution across all domains of life, but is especially pervasive in more complex eukarya. There are various proposals on why duplication is so widespread, with many being complementary and possibly all part of the larger truth. Without being able to replay the evolution of life, it is challenging to examine these reasons in depth. However, the duplication-heteromerisation pathway discussed here aligns well with existing theories [70, 81, 85] and provides an evolutionary advantage outside of selection pressure with minimal assumptions.

Symmetric homomeric interactions form much more quickly but are more susceptible to misassembly than heteromeric interactions, recovering dynamics observed in other more biologically oriented models [42, 43]. Duplication-heteromerisation allows evolution to cherry-pick the best aspects of both types of interaction, transforming fast-forming symmetric homomeric interactions into evolvable heteromeric interactions. The duplication pathway is only indirectly beneficial, by enhancing the evolvability through heteromerisation. This matches the prognosis that duplication by itself doesn't confer an advantage [70].

Duplication accelerates the evolution of larger phenotypes and is a repeatable pathway, as seen in *in silico* simulations. Several qualitative predictions made by the model can be recovered from experimentally recorded proteins in the PDB. Heteromeric interactions have an excessive correlation of domains, i.e. more homologous heteromeric interactions than expected by chance. Homologous heteromeric interactions also robustly have larger buried surface areas than non-homologous ones. Perhaps the strongest piece of evidence is that subunit binding sequence alignment is substantially better for homologous interactions. Together, these results show that duplication is a significant contributor to the observed evolution of complexity in protein complexes.

CHAPTER FIVE

CONCLUSION

Proteins play fundamental roles in innumerable biological processes, and appear in a diverse range of shapes and sizes. The dynamics governing the interactions between protein subunits, or even higher-order interactions between complexes, are influenced by geometric and other external factors. Coarse-graining proteins into a simpler representation allows general characteristics of protein complexes to be modelled and explored. Lattice self-assembly models, conceptually linked to protein quaternary structure, facilitate qualitative predictions and novel insights into protein complex self-assembly.

The research presented in this thesis is progress towards a more general understanding of interactions between protein subunits. Each segment can stand alone, but together they form a coherent whole. Crucially, these results highlight the intricate relationship between immediate assembly dynamics and long-term evolutionary dynamics. In particularly, understanding the dynamics acting in one of these scales contains insight on the other.

Chapter 2 focuses on the continuation and completion of an algorithmic approach to the classification of the boundedness and determinism of an arbitrary set of self-assembling subunits. The revamped classification scheme remedies multiple deficiencies and extends the range of allowed interactions, including self-interactions. This general framework provides a link between self-assembly behaviour and proteopathic risk, where proteins containing repeated patterns (cycles) are typically only a few mutations away from uncontrolled aggregation (unbound).

A comparison between implementations of the previous stochastic assembly classification scheme and the complete assembly graph formalism conclusively shows a material gain in speed. While the stochastic assembly scheme was occasionally incorrect, the new scheme maintains flawless classification. Multiple extensions can also be applied to the assembly graph approach. For example, seed dependence or variable interaction environments can be considered, if relevant to a target system's context. Other mathematical relationships natural to assembly graphs, such as isomorphisms, assist in genotype-phenotype maps currently under research.

Chapter 3 generalises the polyomino model by introducing binary strings as the genotypic elements which encode binding sites. This generalisation greatly expands the scope of the self-assembly model, allowing new dynamics to be studied without sacrificing model tractability. Interaction strength evolution can then naturally be modelled, in parallel with analytic predictions from a suitable Markov chain approximation.

Deterministic phenotypes, which can assemble with any possible ordering of steps, obey these strength approximations, while nondeterministic phenotypes diverge rapidly and nontrivially. Interactions which contribute to nondeterminism, i.e. inherently impose a preferred order to assembly, are driven by selection pressures to strengthen or weaken depending on whether the subunits needed to respectively bind earlier or later in the assembly process. This "core" and "periphery" assembly is ingrained into preferred assembly pathways. Evolutionary pathway also reflect this assembly preference, as the order of the evolutionary transitions corresponds to the optimal ordering of interaction strengths. These qualitative observations match small-scale experimental observations found in the literature [47, 62].

Chapter 4 focuses on a well observed but less well understood genetic event: duplication. In addition, self-interactions are explicitly allowed in the full model unlike in previous polyomino studies. An analytic pathway through duplication-heteromerisation allows evolution to optimally exploit the fast-forming nature of symmetric homomeric interactions as well as the greater evolvability of heteromeric interactions. This pathway is sufficiently repeatable, enabling a much faster increase of structural complexity than possible without duplication.

Multiple qualitative predictions generated by *in silico* simulations have been tested against experimental data taken from the Protein Data Bank. Proteins from the dataset display a significant overabundance of homologous heteromeric interactions, suggesting origins through duplication. Likewise, homologous heteromeric interactions have larger buried surface areas, conceptually similar to binary string binding site length, compared to their non-homologous counterparts. Most conclusively, homologous heteromeric interactions have statistically more confident alignments at the binding residue level when comparing subunit binding regions. Collectively, these results strongly suggest that the duplication of symmetric homomeric interactions play a meaningful role in the evolution of heteromeric protein complexes.

5.1 Avenues for further investigation

As a result of the main findings presented in this thesis, various research offshoots have emerged. These additional avenues likely require an extended period of time to aptly explore and address, and so are described here for future investigations.

Fixed interaction geometry is inherent in the lattice description of self-assembly. However, it does not apply to proteins, which can bind in a wide variety of configurations. Shifting to a more "free-form" concept of subunits could model dynamics that cannot be modelled with fixed interaction geometry. With circular subunits, interactions could occur between any genotypic segments on the perimeter, rather than between four independent and isolated faces. While this approach raises many additional questions on steric complications, it may also provide an opportunity to explore how proteins overcome them.

Similarly, rigid interaction geometry dictates that the size of binding site is fixed. Introducing an ability to handle binding sites of different sizes, possibly through the framework mentioned above, will allow the sizes themselves to evolve. Dynamic binding sites could explore the evolution of interaction strengths from a different angle, exploring how interface size grows alongside structural complexity. In addition, it could be used to examine under what context might interactions evolve to be symmetrically or asymmetrically sized.

General polyomino model studies thus far also do not address many thermodynamic considerations. Probabilistic decoupling of previous bindings establishes new dynamics which may provide additional capacity to explore the evolution and maintenance of interaction strengths. All of these extensions to the model are relatively unexplored, and could provide a way to redress some of the more unrealistic simplifications of proteins.

In addition to proposed studies above, there are several topics that have already been probed. These studies on modular interactions and genetic regulation have been conducted with the integer-labelled polyomino model. The binary string generalisation introduced in Chapter 3 overcomes multiple drawbacks of this previous model, allowing graduated interaction strengths as well as relaxed transitivity, which enables smoother evolutions. As such, these previously unsuccessful studies will almost certainly benefit from the generalised polyomino model.

Modularity, a measure of reusing the same components in multiple circumstances, has been shown to play a role in efficient evolution by Kashtan and Alon [97], amongst others. This was applied to polyominoes by evolving on a fitness landscape which alternated between two phenotypes with a common subset of interactions. However, due the strong risk of deleterious assembly when multiple interactions are repeated, little modularity emerged.

As discussed previously, interactions in the generalised model are not necessarily transitive. With a more complex web of interactions possible, modular structures have a better chance to emerge while remaining bound and mostly deterministic. This newer framework might then be able to replicate earlier claims that modular evolution can outpace independent evolution. Such a result was hinted at in the plasticity of Figure 3.14, but should be examined in a more focused study.

Genetic regulation, loosely the ability to switch on and off parts of a genotype, plays a huge role in the control of genetic transcription and translation of genes into proteins. Regulation can be imposed in the self-assembly model by additional genotypic elements which control whether the subsequent encoded subunit is produced. Abstract but multiscale models have shown promising incorporation of regulation into protein assembly [98]. Early attempts with the polyomino model used randomly mutating binary switches, but they could not adapt to their local fitness landscape.

With binary string binding sites leading to a greater number and diversity of interactions, it would be easier for subunit and regulatory interactions to coexist and potentially co-interact. Recoupling subunit and regulatory interactions through using the generalised model may reveal the spontaneous emergence of regulatory networks.

5.2 Final thoughts

Modelling the self-assembly of protein quaternary structure as polyominoes leaves a fair amount to be desired. Protein conformation and cooperative binding dynamics are disregarded, together with other important specifics of protein structures and their interactions. However, through these simplified approaches, we can gain insights that are otherwise difficult to model, precisely because of the details that are omitted here. We finally *can* see the wood for the (approximated) trees.

The polyomino model has already provided additional reasoning on the risk of protein aggregation, asymmetric interaction strengths, and why duplication may be so potent. While there are plenty of dynamics that cannot be explored using this model, there are many left that can. Hopefully this model will continue to serve a role in understanding the general nature of protein interactions and their resultant complexes.

REFERENCES

[1] D Wirtz, K Konstantopoulos, and PC Searson. “The physics of cancer: the role of physical interactions and mechanical forces in metastasis”. *Nature Reviews Cancer* 11.7 (2011), p. 512.

[2] DW Sims et al. “Scaling laws of marine predator search behaviour”. *Nature* 451.7182 (2008), p. 1098.

[3] MK Transtrum et al. “Perspective: Sloppiness and emergent theories in physics, biology, and beyond”. *The Journal of chemical physics* 143.1 (2015).

[4] GEP Box and NR Draper. *Empirical model-building and response surfaces.* John Wiley & Sons, 1987.

[5] K Christensen, KA Manani, and NS Peters. “Simple model for identifying critical regions in atrial fibrillation”. *Physical review letters* 114.2 (2015), p. 028104.

[6] D Helbing et al. “Saving human lives: What complexity science and information systems can contribute”. *Journal of statistical physics* 158.3 (2015), pp. 735–781.

[7] E Ising. “Beitrag zur theorie des ferromagnetismus”. *Zeitschrift für Physik A Hadrons and Nuclei* 31.1 (1925), pp. 253–258.

[8] M Kouza et al. “The GOR method of protein secondary structure prediction and its application as a protein aggregation prediction tool”. *Prediction of Protein Secondary Structure*. Springer, 2017, pp. 7–24.

[9] KA Dill. “Theory for the folding and stability of globular proteins”. *Biochemistry* 24.6 (1985), pp. 1501–1509.

[10] SE Ahnert et al. “Principles of assembly reveal a periodic table of protein complexes”. *Science* 350.6266 (2015), aaa2245.

[11] C Empereur-Mot et al. “Proteins evolve on the edge of supramolecular self-assembly”. *Nature* 548.7666 (2017), p. 244.

[12] P Sartori and Sf Leibler. “Lessons from equilibrium statistical physics regarding the assembly of protein complexes”. *Proceedings of the National Academy of Sciences* (2019).

[13] P Tompa and M Fuxreiter. “Fuzzy complexes: polymorphism and structural disorder in protein–protein interactions”. *Trends in biochemical sciences* 33.1 (2008), pp. 2–8.

[14] H Wang. “Proving theorems by pattern recognition—II”. *Bell system technical journal* 40.1 (1961), pp. 1–41.

[15] CH Bennett. “Logical reversibility of computation”. *IBM journal of Research and Development* 17.6 (1973), pp. 525–532.

[16] E Winfree. "Algorithmic self-assembly of DNA". PhD dissertation. California Institute of Technology, 1998.

[17] D Woods et al. "Diverse and robust molecular algorithms using reprogrammable DNA self-assembly". *Nature* 567.7748 (2019), p. 366.

[18] CG Evans and E Winfree. "Physical principles for DNA tile self-assembly". *Chemical Society Reviews* 46.12 (2017), pp. 3808–3829.

[19] SE Ahnert et al. "Self-assembly, modularity, and physical complexity". *Physical Review E* 82.2 (2010), p. 026117.

[20] IG Johnston et al. "Evolutionary dynamics in a simple model of self-assembly". *Physical Review E* 83.6 (2011), p. 066105.

[21] SF Greenbury et al. "A tractable genotype–phenotype map modelling the self-assembly of protein quaternary structure". *Journal of The Royal Society Interface* 11.95 (2014), p. 20140249.

[22] SF Greenbury and SE Ahnert. "The organization of biological sequences into constrained and unconstrained parts determines fundamental properties of genotype–phenotype maps". *Journal of The Royal Society Interface* 12.113 (2015), p. 20150724.

[23] SF Greenbury et al. "Genetic correlations greatly increase mutational robustness and can both reduce and enhance evolvability". *PLoS computational biology* 12.3 (2016), e1004773.

[24] S Tesoro, SE Ahnert, and AS Leonard. "Determinism and boundedness of self-assembling structures". *Physical Review E* 98.2 (2018), p. 022113.

[25] S Tesoro. "Non-deterministic self-assembly on a two-dimensional lattice". PhD dissertation. University of Cambridge, 2017.

[26] C Lin et al. "DNA tile based self-assembly: building complex nanoarchitectures". *ChemPhysChem* 7.8 (2006), pp. 1641–1647.

[27] F Hong et al. "DNA origami: scaffolds for creating higher order structures". *Chemical reviews* 117.20 (2017), pp. 12584–12640.

[28] NC Seeman and HF Sleiman. "DNA nanotechnology". *Nature Reviews Materials* 3.1 (2017), pp. 1–23.

[29] Y Bai, Q Luo, and J Liu. "Protein self-assembly via supramolecular strategies". *Chemical Society Reviews* 45.10 (2016), pp. 2756–2767.

[30] JS Valastyan and S Lindquist. "Mechanisms of protein-folding diseases at a glance". *Disease models & mechanisms* 7.1 (2014), pp. 9–14.

[31] F Chiti and CM Dobson. "Protein misfolding, amyloid formation, and human disease: a summary of progress over the last decade". *Annual review of biochemistry* 86 (2017), pp. 27–68.

[32] R Rizzi. "Minimum weakly fundamental cycle bases are hard to find". *Algorithmica* 53.3 (2009), pp. 402–424.

[33] NM Taleb. *The black swan: The impact of the highly improbable.* Vol. 2. Random house, 2007.

[34] PJ Cameron. "Automorphisms of graphs". *Topics in algebraic graph theory* 102 (2004), pp. 137–155.

[35] HA Helfgott, J Bajpai, and D Dona. *Graph isomorphisms in quasi-polynomial time.* 2017. arXiv: `1710.04574 [math.GR]`.

[36] SE Ahnert. “Structural properties of genotype–phenotype maps”. *Journal of The Royal Society Interface* 14.132 (2017), p. 20170275.

[37] F Ruskey, C Savage, and TMY Wang. “Generating necklaces”. *Journal of Algorithms* 13.3 (1992), pp. 414–430.

[38] AM Turing. “On computable numbers, with an application to the Entscheidungsproblem”. *Proceedings of the London mathematical society* 2.1 (1937), pp. 230–265.

[39] P-É Meunier and D Woods. “The non-cooperative tile assembly model is not intrinsically universal or capable of bounded Turing machine simulation”. *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing.* 2017, pp. 328–341.

[40] J Hendricks et al. “The power of duples (in self-assembly): It’s not so hip to be square”. *Theoretical Computer Science* 743 (2018), pp. 148–166.

[41] AS Leonard and SE Ahnert. “Evolution of interface binding strengths in simplified model of protein quaternary structure”. *PLoS computational biology* 15.6 (2019), e1006886.

[42] DB Lukatsky et al. “Structural similarity enhances interaction propensity of proteins”. *Journal of molecular biology* 365.5 (2007), pp. 1596–1606.

[43] DB Lukatsky and EI Shakhnovich. “Statistically enhanced promiscuity of structurally correlated patterns”. *Physical Review E* 77.2 (2008), p. 020901.

[44] JR Brender and Y Zhang. “Predicting the effect of mutations on protein-protein binding interactions through structure-based interface profiles”. *PLoS computational biology* 11.10 (2015), e1004494.

[45] MA Siddiq, GKA Hochberg, and JW Thornton. “Evolution of protein specificity: insights from ancestral protein reconstruction”. *Current opinion in structural biology* 47 (2017), pp. 113–122.

[46] JA Marsh et al. “Structural and evolutionary versatility in protein complexes with uneven stoichiometry”. *Nature communications* 6 (2015), p. 6394.

[47] ED Levy et al. “Assembly reflects evolution of protein complexes”. *Nature* 453.7199 (2008), p. 1262.

[48] JA Marsh et al. “Protein complexes are under evolutionary selection to assemble via ordered pathways”. *Cell* 153.2 (2013), pp. 461–470.

[49] JR Ellis. “Protein misassembly”. *Molecular Aspects of the Stress Response: Chaperones, Membranes and Networks.* Springer, 2007, pp. 1–13.

[50] A Lopes et al. “Protein-protein interactions in a crowded environment: an analysis via cross-docking simulations and evolutionary information”. *PLoS computational biology* 9.12 (2013).

[51] E Laine and A Carbone. “Protein social behavior makes a stronger signal for partner identification than surface geometry”. *Proteins: Structure, Function, and Bioinformatics* 85.1 (2017), pp. 137–154.

[52] AL Goldberg and KL Rock. “Proteolysis, proteasomes and antigen presentation”. *Nature* 357.6377 (1992), pp. 375–379.

[53] SW Englander, L Mayne, and MMG Krishna. “Protein folding and misfolding: mechanism and principles”. *Quarterly reviews of biophysics* 40.4 (2007), pp. 1–41.

[54] D Caballero et al. “Steric interactions determine side-chain conformations in protein cores”. *Protein Engineering, Design and Selection* 29.9 (2016), pp. 367–376.

[55] DA Levin and Y Peres. *Markov chains and mixing times*. Vol. 107. American Mathematical Soc., 2017.

[56] S Schaper and AA Louis. "The arrival of the frequent: how bias in genotype-phenotype maps can steer populations to local optima". *PloS one* 9.2 (2014), e86635.

[57] SG Foy et al. "A shift in aggregation avoidance strategy marks a long-term direction to protein evolution". *Genetics* 211.4 (2019), pp. 1345–1355.

[58] L Agozzino and KA Dill. "Protein evolution speed depends on its stability and abundance and on chaperone concentrations". *Proceedings of the National Academy of Sciences* 115.37 (2018), pp. 9092–9097.

[59] E Natan et al. "Cotranslational protein assembly imposes evolutionary constraints on homomeric proteins". *Nature structural & molecular biology* 25.3 (2018), p. 279.

[60] LX Peterson et al. "Modeling the assembly order of multimeric heteroprotein complexes". *PLoS computational biology* 14.1 (2018), e1005937.

[61] M Bertoni et al. "Modeling protein quaternary structure of homo-and hetero-oligomers beyond binary interactions by homology". *Scientific reports* 7.1 (2017), p. 10480.

[62] JA Marsh and SA Teichmann. "Structure, dynamics, assembly, and evolution of protein complexes". *Annual review of biochemistry* 84 (2015), pp. 551–575.

[63] A Sharir-Ivry and Y Xia. "The impact of native state switching on protein sequence evolution". *Molecular biology and evolution* 34.6 (2017), pp. 1378–1390.

[64] T Perica, C Chothia, and SA Teichmann. "Evolution of oligomeric state through geometric coupling of protein interfaces". *Proceedings of the National Academy of Sciences* 109.21 (2012), pp. 8127–8132.

[65] D Kleiner et al. "The interdimeric interface controls function and stability of Ureaplasma urealiticum methionine S-adenosyltransferase". *Journal of molecular biology* 431.24 (2019), pp. 4796–4816.

[66] G Abrusán and JA Marsh. "Ligand binding site structure influences the evolution of protein complex function and topology". *Cell reports* 22.12 (2018), pp. 3265–3276.

[67] RJ Ellis. "Assembly chaperones: a perspective". *Philosophical Transactions of the Royal Society B: Biological Sciences* 368.1617 (2013), p. 20110398.

[68] AS Leonard and SE Ahnert. "Duplication accelerates the evolution of structural complexity in protein quaternary structure". *bioRxiv* (2020).

[69] AD McLachlan. "Gene duplications in the structural evolution of chymotrypsin". *Journal of molecular biology* 128.1 (1979), pp. 49–79.

[70] M Kimura. "The neutral theory of molecular evolution: a review of recent evidence". *The Japanese Journal of Genetics* 66.4 (1991), pp. 367–386.

[71] M Lynch and JS Conery. "The origins of genome complexity". *science* 302.5649 (2003), pp. 1401–1404.

[72] S Magadum et al. "Gene duplication as a major force in evolution". *Journal of genetics* 92.1 (2013), pp. 155–161.

[73] TA Gibson and DS Goldberg. "Questioning the ubiquity of neofunctionalization". *PLoS computational biology* 5.1 (2009), e1000252.

[74] KD Crow and GP Wagner. "What is the role of genome duplication in the evolution of complexity and diversity?" *Molecular biology and evolution* 23.5 (2005), pp. 887–892.

[75] TJ Treangen and EPC Rocha. "Horizontal transfer, not duplication, drives the expansion of protein families in prokaryotes". *PLoS genetics* 7.1 (2011), e1001284.

[76] EC Tromer et al. "Mosaic origin of the eukaryotic kinetochore". *Proceedings of the National Academy of Sciences* 116.26 (2019), pp. 12873–12882.

[77] R Dandage and CR Landry. "Paralog dependency indirectly affects the robustness of human cells". *Molecular Systems Biology* 15.9 (2019), e8871.

[78] G Diss et al. "Gene duplication can impart fragility, not robustness, in the yeast protein interaction network". *Science* 355.6325 (2017), pp. 630–634.

[79] KA Cannon, JM Ochoa, and TO Yeates. "High-symmetry protein assemblies: Patterns and emerging applications". *Current opinion in structural biology* 55 (2019), pp. 77–84.

[80] D Myers-Turnbull et al. "Systematic detection of internal symmetry in proteins using CE-Symm". *Journal of molecular biology* 426.11 (2014), pp. 2255–2268.

[81] A Force et al. "Preservation of duplicate genes by complementary, degenerative mutations". *Genetics* 151.4 (1999), pp. 1531–1545.

[82] M Clemente-Ruiz et al. "Gene dosage imbalance contributes to chromosomal instability-induced tumorigenesis". *Developmental cell* 36.3 (2016), pp. 290–302.

[83] PP Edger and JC Pires. "Gene and genome duplications: the impact of dosage-sensitivity on the fate of nuclear genes". *Chromosome Research* 17.5 (2009), p. 699.

[84] AN Kolmogorov. "On tables of random numbers". *Theoretical Computer Science* 207.2 (1998), pp. 387–395.

[85] S Ohno. *Evolution by gene duplication.* Springer-Verlag, 1970.

[86] wwPDB consortium. "Protein Data Bank: the single global archive for 3D macromolecular structure data". *Nucleic acids research* 47.D1 (2018), pp. D520–D528.

[87] DB Wetlaufer. "Nucleation, rapid folding, and globular intrachain regions in proteins". *Proceedings of the National Academy of Sciences* 70.3 (1973), pp. 697–701.

[88] NL Dawson et al. "CATH: an expanded resource to predict protein function through structure and sequence". *Nucleic acids research* 45.D1 (2016), pp. D289–D295.

[89] J Chen, N Sawyer, and L Regan. "Protein–protein interactions: General trends in the relationship between binding affinity and interfacial buried surface area". *Protein Science* 22.4 (2013), pp. 510–515.

[90] E Krissinel and K Henrick. "Inference of macromolecular assemblies from crystalline state". *Journal of molecular biology* 372.3 (2007), pp. 774–797.

[91] F Madeira et al. "The EMBL-EBI search and sequence analysis tools APIs in 2019". *Nucleic acids research* (2019).

[92] J Tong et al. "Using homology relations within a database markedly boosts protein sequence similarity search". *Proceedings of the National Academy of Sciences* 112.22 (2015), pp. 7003–7008.

[93] NJ Marianayagam, M Sunde, and JM Matthews. "The power of two: protein dimerization in biology". *Trends in biochemical sciences* 29.11 (2004), pp. 618–625.

[94] JB Pereira-Leal et al. "Evolution of protein complexes by duplication of homomeric interactions". *Genome biology* 8.4 (2007), R51.

[95] JA Marsh and SA Teichmann. "Protein flexibility facilitates quaternary structure assembly and evolution". *PLoS biology* 12.5 (2014), e1001870.

[96] WR Pearson. “An introduction to sequence similarity (“homology”) searching”. *Current protocols in bioinformatics* 42.1 (2013), pp. 3–1.

[97] N Kashtan and U Alon. “Spontaneous evolution of modularity and network motifs”. *Proceedings of the National Academy of Sciences* 102.39 (2005), pp. 13773–13778.

[98] CF Arias et al. “toyLIFE: a computational framework to study the multi-level organisation of the genotype-phenotype map”. *Scientific reports* 4 (2014), p. 7549.

APPENDIX

A

SOFTWARE AND SIMULATION DETAILS

Pertinent details to evolutionary dynamics or algorithm implementations are provided in the main text. However, there are additional, ancillary details provided here regarding lesser simulation parameterisations. A core theme of this work is that the dynamics generally transcend any specific parameterisation. For completeness though, all parameters are detailed below.

A.1 Simulation parameters

Most units of presented results were specialised on slightly different aspects of evolutionary dynamics, and correspondingly needed slightly modified simulations. The common parameters, as well as those unique to certain simulations, are discussed below.

One of the key results repeatedly emphasised in the main text was the generality of observations. Most parameters took on arbitrary values, chosen through initial estimation or adjusted in response to simulated data. Their final values were primarily chosen to balance clarity of results and the limitations of computational power.

In all simulations, the population size was fixed at 100 individuals. This value was large enough to minimise stochastic fluctuations in selection dynamics and risk of extinction, but maintain the ability to collect data over many generations. Other common parameters, which varied for different simulation were the number of generations G, the mutation rate μ, and the previously stated repeated number of assemblies K. There is also the nondeterminism threshold χ, which is the minimum fraction of assemblies required to be a valid phenotype.

The binding strength evolution dynamics shown in Chapter 3 used parameters shown in Table A.1. Importantly, these simulations used a fitness function relating to assembly graph edges rather than phenotype size.

Duplication simulations from Chapter 4 contained mostly similar parameters, except the new addition of duplication and deletion, given by δ. These rates were set by the same parameter to ensure approximately stable genotype size, rather than growing or shrinking

Table A.1: All assembly graphs were fixed at two subunits, as indicated by the given subset in the main text. Binding site lengths were $L = 64$ with a critical strength of $\hat{S}_c = 0.6875$. Subunit + orientation information was used, again indicated by the coloured assembly tiles. The dynamic landscape evolutions rewarded fitness alternatively to the 12-mer and 10-mer with a period of $\Omega = 50$, and fitness was the weighted sum for all polyominoes, hence $\chi = 0, \gamma = 1$.

	Phenotype relation		Simulation parameters					
System	Polyomino	Composition	K	χ	μ	G	γ	T
$\langle \hat{S} \rangle$ evolution	free	S+O	25	1/3	$0.125/L$	2000	5	25
Dynamic	free	S+O	150	0	$0.125/L$	750	1	10

too rapidly. These simulations returned to the standard size based fitness function. In addition, these simulations required less phylogenetic reconstruction, and hence could simulate more generations. Simulations generally terminated early, as they were set with a three stable interaction limit.

Table A.2: Simulations varied for L and $\hat{S}_c$, with values explicitly provided in relevant figures. All simulations had variable genotype length, with a dynamically regenerating neutral space of three subunits. Anytime an interaction formed with a neutral subunit, a new randomly generated neutral subunit was added to the end of the genotype.

	Phenotype relation		Simulation parameters				
System	Polyomino	Composition	K	χ	μ	G	δ
Duplication	free	shape	20	1/3	$0.25/L$	10^6	0.05

A.2 Code availability

All of software used in producing the data and analyses presented in the main text are publicly available online. Each have flexible licenses, designed to easily allow modification and repurposing provided the new uses are also freely available. Projects are not all actively maintained, but should be usable with currently available software versions.

The assembly graph classification scheme discussed in Chapter 2 is implemented at `https://github.com/ASLeonard/SLAM`, titled **S**quare **L**attice **A**ssembly **M**odel (SLAM). Although the assembly graph classification scheme can be most generally implemented in terms of edges, this program is optimised to use integer labels directly. There are additionally various scripts that can animate polyomino assembly according to different implementation models.

Many individual aspects of polyomino models were subsumed by assembly graphs, such as the stochastic assembly algorithm. Provided a unique model had a description for forming an assembly graph, the subsequent assembly took on a global form. As such, a generic framework was created to allow rapid prototyping of different genotypic descriptions for new models. The core of this framework is implemented at `https://github.com/ASLeonard/polyomino_core`.

The generalised polyomino model extends this framework by providing the binary string binding site genotype elements. This was initially implemented specifically for the evolution of binding strengths in Chapter 3, but was later adopted in multiple projects. The initial implementation is found at `https://github.com/ASLeonard/polyomino_interfaces`, where binary strings are restricted to fixed width integers (8, 16, 32, and 64 bits, with experimental support for 128 bits).

A more flexible model was used for Chapter 4, allowing arbitrary length binary string binding sites. This model is found at `https://github.com/ASLeonard/duplication`. The PDB analysis was initially modified code provided by Ahnert, and later extended into SuBSeA. The ongoing development of SuBSeA can be found at `https://github.com/ASLeonard/SuBSeA`.

The general framework laid out above has also been used in studies beyond those presented here. The briefly discussed advanced genotype-phenotype sampling method from Chapter 2, central to forthcoming work by Jouffrey, Leonard, and Ahnert, has been extended and implemented at `https://github.com/vatj/gpmap_integer_polyomino`.

A.3 Pseudocode implementations

Although all the code used across this work is freely available as described above, several of the key algorithms are expressed below. In particular, the assembly algorithm is outlined as introduced in the text, as exact assembly implementation has been demonstrated to have consequences.

The assembly algorithm is outlined below in Algorithm 1. This general form of edges and weights removes the need to re-implement a polyomino assembly algorithm for every new form of genotypic element and interaction matrix. Similarly, other forms of unbound threshold can be used, if larger or smaller assemblies are desired, or just pure spatial constraints.

Algorithm 1: Polyomino assembly overview

```
Input: assembly graph edges $E$
Result: polyomino structure $\mathbf{P}$
initiate structure $\mathbf{P}$ with seed                       // random or fixed seed
identify possible_bindings interactions using $E$
while possible_bindings is not empty:
    new lattice binding site $b_{ij}$ = CHOICE(possible_bindings)
    /* can weight CHOICE if $E$ contains interaction probabilities */
    $\mathbf{P} \leftarrow b_{ij}$                                  // add new tile to structure
    ; remove $b_{ij}$ from possible_bindings
    update possible_bindings for new interactions to $b_i$
    if SIZE ($\mathbf{P}$) > $4N_s^2$:                         // Upper bound for $N_s$ subunits
        return unbound
return $\mathbf{P}$
```

Similarly, phenotype comparison is crucial to phylogenetic reconstruction and many other population tracking features used to generate results. An algorithmic overview is provided below in Algorithm 2, and more complete details on the composition relation $\mathcal{I}$ and mapping strategy $\mathcal{M}$ can be found in Appendix B. This process provides objective representations of phenotypes, so different model simulations at different times will produce equivalent numeric strings.

Algorithm 2: Phenotype representation overview

```
Input: polyomino structure $\mathbf{P}$
Parameters: polyomino composition relation $\mathcal{I}$
Result: invariant phenotype representation $\mathcal{P}$
$\Delta X, \Delta Y = \text{extent}(\mathbf{P})$                        // spatial width and height
fill lattice details $\mathbf{L} = \mathcal{I}(\mathbf{P})$
if $\Delta Y > \Delta X$:                                              // convention choice
    rotate clockwise $\pi/2$
/* Next step depends on polyomino definition and inherent symmetry of $\mathbf{P}$ */
for each unique symmetry operations $\hat{O}$:
    apply $\mathbf{L}' = \hat{O}(\mathbf{L})$                          // e.g. rotate or reflect L
    relabel $\mathbf{L}'$ with mapping strategy $\mathcal{M}$ append relabelled $\mathbf{L}'$ to minimals list
return MIN (minimals)                                                  // lexicographic sort
```

Simulating evolution is fairly straightforward, just repeatedly mutating and selecting individuals as shown in Algorithm 3. The same general form is used in all simulations, allowing a high degree of reusability. Specific functions can be inserted into the framework, normally after assembly, which then captures any desired dynamics, e.g. interaction strengths, sequence similarity, etc.

Algorithm 3: Evolution framework

```
Parameters: fitness function $\mathcal{F}$
Result: evolutionary history
Initialise genotype population $\mathcal{G}$                   // randomly or with fixed sequence
for $k \to k_{\text{max}}$:                                     // iterate for $k_{\text{max}}$ generations
    for genotype $g \in \mathcal{G}$:
        mutate $g$ given rate $\mu$
        determine phenotype $\mathcal{P} = \text{ASSEMBLE}(g)$
        /* extract any simulation specific information */
        population_fitnesses $\leftarrow \mathcal{F}(g, \mathcal{P})$   // assess individual $g$'s fitness
    $\mathcal{G}' = \text{CHOICES}$ (population_fitnesses)
                                                                // roulette wheel selection
    repopulate new generation $\mathcal{G} \leftarrow \mathcal{G}'$
return evolutionary history
```

APPENDIX

B

POLYOMINO COMPARISON

Two-dimensional lattice self-assembly results in polyominoes, sets of continuously connected square tiles. Determining if two self-assembling structures are equivalent can be highly nontrivial. Different polyomino definitions allow or forbid certain symmetries, while supplemental information from the assembly process can also be meaningful.

Polyomino comparison determines if two assemblies are equivalent and correspond to a single phenotype. This occurs in two distinct situations. The first is comparing repeated assemblies from the same genotype, either with fixed or random seeds. This case is fairly simple, as if the assemblies are the same, they can only differ up to some easily detectable absolute rotation. The second situation is comparing assemblies from different genotypes. This is much harder, as equivalent assemblies are not necessarily separated by just symmetry operations. An invariant representation, introduced below, connects the desired properties of genotypes to observable properties of phenotypes.

B.1 Assembly structure details

There are two levels on which structural information is captured during self-assembly: macroscopic shape and microscopic detail. Macroscopic shape is the polyomino, determined by the configuration of tiles glued together, while microscopic details cover the subunit type and orientation. Depending on the context of the investigation, as much or little of this information can be used to determine what stipulates the equivalence relation.

B.1.1 Polyomino definitions

Polyominoes can be defined under three main levels, with decreasing strictness: fixed, one-sided, and free. Shapes which are rotations of each other are distinct under fixed polyominoes, but are not under one-sided. Shapes which are reflections of each other are distinct under one-sided, but are not under free. There is no “correct” choice of definition, but rather it should be selected to best match the constraints and considerations of the target system under study.

For a fixed polyomino, shapes can be compared directly, while for one-sided, there are four potential orientations to check for equivalency. Finally, for a free polyomino, there are four rotations for each chirality, making a total of eight potential orientations. Not all of these will be distinct, i.e. a two-by-two polyomino may have C_4 rotational symmetry, but distinct reflections, and so only two unique configurations to check.

B.1.2 Structure composition

In addition to the hierarchical polyomino definitions, the composition of the assembly can be used to distinguish identical shapes with different tiers of strictness. If composition does not contribute meaningfully, all subunits are treated equally, and comparison acts on standard polyominoes. However, if it does play a role, e.g. a homodimer and heterodimer have different functions or properties, this information can "adorn" the polyominoes.

Supplementing polyominoes with composition information breaks the symmetry between genotype and phenotype. All polyomino studies, excluding fixed-seed considerations, do not impose any special conditions that the genotype order matters, e.g. two subunits can swap order with no observable effect. This meaningless swap would change the composition information of the assembly, as the subunit types have swapped as well. To correct this, composition information must be representable in a manner which does not depend on any specific genotype order. Any isomorphic assembly graph is, by definition, equivalent, and so must be mappable to a single composition.

B.2 Polyomino encoding

Polyominoes can be characterised as numeric strings to facilitate comparison. First, an r-tile tall and c-tile wide polyomino is bounded by an unpadded box, such that the lattice shape matches an $r \times c$ matrix. Each element in the matrix is concatenated into a numeric string, from top left to bottom right, moving row-wise.

Occupied lattice sites are labelled according to the chosen comparison relation, as a function of the placed subunit type and rotation, $\mathcal{I}\left(T_i, \theta_i\right)$ that were used during assembly. Empty lattice sites are denoted by 0. The three currently used labelling systems are

- Shape only: $\mathcal{I}\left(T_i, \theta_i\right) = 1$
- Subunit: $\mathcal{I}\left(T_i, \theta_i\right) = T_i$
- Subunit + orientation: $\mathcal{I}\left(T_i, \theta_i\right) = 4 \cdot T_i + \theta_i - 3$

By convention, subunit types start from 1, while θ counts the number of $\pi/2$ clockwise rotations from the vertical, starting from 0. Additional labelling systems can easily be constructed, depending on what information is pertinent to the model. An example of subunit + orientation labelling is shown in Figure B.1. Although numeric strings generated in this fashion are straightforward to compare, they still do not possess the desired genotype symmetries.

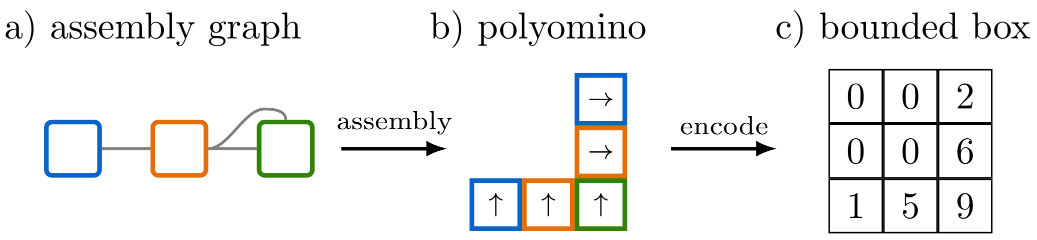


Figure B.1: a) An assembly graph with three subunits and three interactions, using the same colour scheme as Figure 1.4. b) The assembly polyomino is a backwards 'L'-shape, where the arrows indicate the direction of placed tiles. c) This polyomino requires a three by three bounding box with four empty elements. Occupied sites are labelled using the subunit + orientation relation, forming the string "0 0 2 0 0 6 1 5 9".

B.2.1 Minimal representation

Desired phenotype symmetries are restored in polyominoes by removing the specific genotype derived information. Relabelling the numeric string achieves this, under a consistently constructed map $\mathcal{M}\colon \mathcal{I} \to \mathcal{I}'$. The first occupied site is assigned the smallest possible label, with $T_1' = 1, \theta_1' = 0$. Other rotations of the same subunit are correspondingly labelled, according to

$$\mathcal{I}'\left(T_i, (\theta_i + n) \pmod 4\right) = \mathcal{I}\left(T_1', \theta_1' + n\right)$$

for $n \in [1, 3]$. After these changes propagate throughout the entire string, the process moves to the next occupied site which has not been relabelled. This site is assigned $T_j' = 2, \theta_j' = 0$, and so on, until all sites have been relabelled. An example of relabelling a numeric string is shown in Figure B.2, for the polyomino in the previous figure.

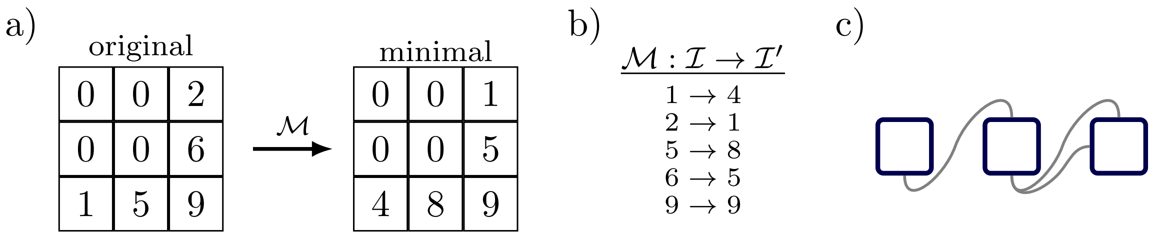


Figure B.2: a) The directly encoded numeric representation is relabelled according to the mapping strategy, $\mathcal{M}$. The exact mapping is shown in b), where the mapping is constructed sequentially, as discussed in the text. c) Occasionally, the direct encoding can be the minimal representation, as for this assembly graph. Notably, this assembly graph is isomorphic to that in Figure B.1, finally connecting genotype and phenotype equivalence through relabelling.

B.3 Lexicographical comparison

Two numeric strings can easily be ordered lexicographically. If every element of the polyominoes' strings were single digits, the comparison would correspond to the size of the number, e.g. the smaller number is the smaller representation. For larger structures potentially containing multiple-digit labels, the comparison relation must proceed one element at a time. Trivially, if the dimensions of the bound box are different, or contain different numbers of nonzero elements, the two polyominoes cannot be equal.

Given two distinct numeric strings $\underline{A}$ and $\underline{B}$ ($\underline{A} \neq \underline{B}$), $\underline{A}$ is more minimal than $\underline{B}$ if the earliest larger element (compared pair-wise) is found in $\underline{B}$. More formally, this can be expressed as

$$\exists \underset{k}{\operatorname{argmin}}(A_k < B_k) \quad \text{S.T.} \quad (A_i \leq B_i)\forall i \in \mathbb{N}^{[1,k)}$$

Depending on the polyomino definition used and a specific polyomino's symmetry, there can be anywhere from one to eight possible rotations/reflections to test. Relabelling should occur after each symmetry permutation, with reflections implemented before rotations. This set of symmetry-related numeric strings is then compared lexicographically. The best representation for a polyomino is thus taken as the smallest minimal encoding, as shown in Figure B.3. Any polyomino represented this way will always equal a similarly processed equivalent assembly, regardless of any genotypic differences.

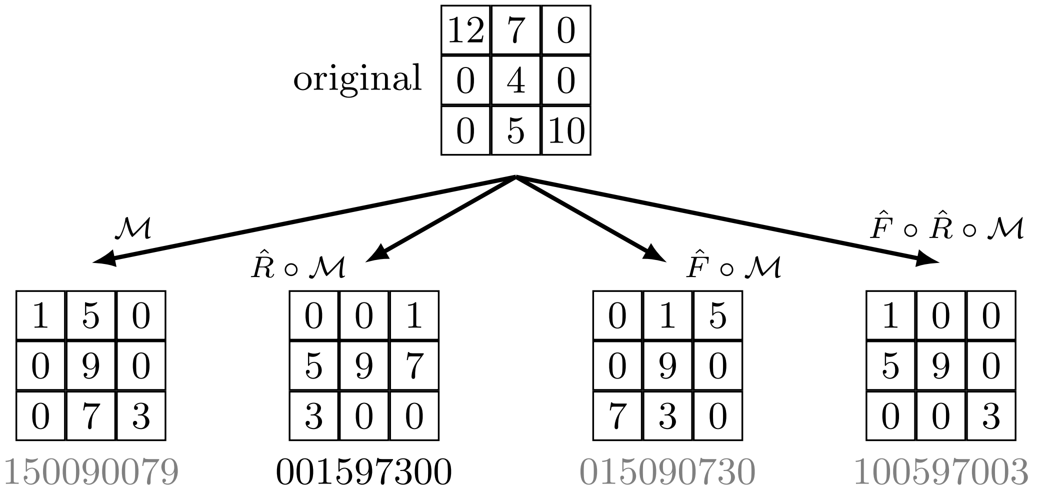


Figure B.3: The original polyomino encoding can be minimised, rotated and minimised, flipped and minimised, or flipped and rotated and minimised. This polyomino has C_2 symmetry, so only 2 rotations and the flipped versions are necessary to check. The second case, rotated and minimised, is the fundamental representation after lexicographical comparisons, with other cases faded out.

Note that this process is not always necessary. For example, if the metric of interest is the population-averaged size, phenotypic equivalence is not necessary. However, once phenotypes need to be tracked over evolving populations with multiple genotypes, then this process becomes mandatory to appropriately group individuals.

APPENDIX

C

PHASE SPACE TRANSITION SUCCESSES

The generality of evolutionary dynamics and phenotype transitions are extensively discussed in Chapter 3. This appendix provides additional details that underpin these dynamics, complementing the bigger picture painted in the main text.

Tractability is a great strength of the generalised polyomino model, but many derivations apply to equilibrium or population trends. Simulations are inherently stochastic, at the levels of both population and assembly dynamics. As such, the following derivations are valid, but should be interpreted cautiously when comparing against individual numerical results. With this in mind, the explanations of observed transition successes are expounded further below, along with approximations for the observed nondeterministic equilbria.

C.1 Direct transitions

Predicting if a phenotype transition will fixate is—on average—straightforward. If there is a net increase in fitness for an individual, it is more likely to reproduce than any other member of the population, and thus eventually dominate the population. Finite populations introduce stochasticity, where fitter individuals are sometimes lost to drift, but in general the above statement is true. As such, determining if a transition leads to an increase in fitness is generally equivalent to determining if the transition is successful.

Fitness depends on multiple terms involving different parameters. There is the proportional fitness increase due to complexity, which can be represented generally as F. In the main text, F = 2, such that each transition doubles the fitness relative to the ancestor. There is also the nondeterminism penalty γ, and the fraction of correct assemblies $\phi = \phi\left(T, \underline{\hat{S}}\right)$, which depends on the temperature parameter and the relevant interaction strengths.

Phase spaces provide a general form for ϕ, such as for the heterotetramer in Figure 3.9. Exponentiating ϕ with γ provides the total penalty applied to nondeterministic assemblies. Multiplying the fitness factor by this total penalty yields the final fitness of some individual. These stages are outlined in Figure C.1, demonstrating the effect of different parameters.

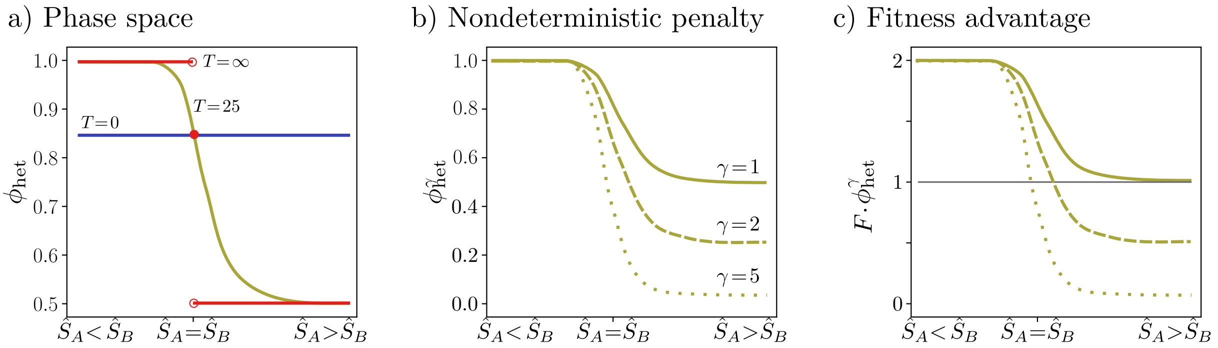


Figure C.1: The heterotetramer phase space depends on intra-subunit and inter-subunit interaction strengths, $\hat{S}_A$ and $\hat{S}_B$ respectively. a) Increasing T, i.e. making the binding probability p_b more sensitive to $\hat{S}$, leads to steeper determinism gradients. In the extreme limit, the fraction of assemblies producing the heterotetramer, ϕ_{het}, becomes a sharp transition. b) Increasing the nondeterminism penalty γ further modulates the curves, only shown for $T = 25$ for simplicity. c) Adding the proportional phenotype fitness factor F, which is two in the simulations, gives the expected fitness post-transition. Anything below the grey line at unit value would not be expected to survive.

Crucially, if any curve in the rightmost panel of Figure C.1 is above consistently above the grey line, then any transition is expected to survive. Final fitness values much greater than one are more likely to overcome stochastic drift and fixate, but any values above one should have moderate transition success rates. Conversely, if any part of a curve is below one, then these transitions would be expected to outright fail in those regions. For example, for $T = 25$ and $\gamma = 2$ or $\gamma = 5$ in the region $\hat{S}_A > \hat{S}_B$, transitions are less fit than their ancestors and should have maximum failure rates.

The above relationship between the product of the fitness factor and ϕ being greater than one can be inverted. The fitness factor can be picked such that some transition is expected to occur, choosing the value $F > \phi^{-\gamma}$. Different phenotypes generally will have different ϕ, such that a constant fitness factor may allow some transitions and not others. Only transitions which satisfy that condition will succeed. Most transitions in Figure 3.12 satisfy this condition, excluding the heterotetramer → 12-mer pathway. The homotetramer → heterotetramer and octomer → 12-mer transitions have lower successes because of a smaller fitness advantage ($F \gtrsim \phi^{-\gamma}$), which is easier to lose through stochastic drift.

C.1.1 Selection driven steady-state

The transition criterion can also be adapted to partially explain the steady-state distribution of strengths driven by nondeterministic selection pressures. As noted in the main text, the ideal steady-state would involve maximally strong and weak interaction. However, these states are highly unstable, and so evolution optimises a balance. This balance was shown to depend on different parameters in Figure 3.8, such as T and γ.

Fitness is weighted by the fraction of correct assemblies ϕ, so maximising this quantity is crucial. However, only a limited number of assemblies are built, typically repeated $K = 10$ times. As such, interaction strengths only need to be optimised such that the chance of any misassembly out of K attempts is minimal. This relates to the binomial distribution, as $B\left(K, \phi^{\gamma}(T)\right)$. The effect of K on the steady-state distributions is much smaller than T or γ. Still, increasing K increases the observed strength gaps, as there are more opportunities to lose fitness and so assemblies have more pressure to avoid such a situation.

C.2 Indirect transitions

The direct transition analysis is conducted using well-defined measures such as the determinism fraction ϕ or expected interaction strengths. However, interaction strengths are per individual, and populations will have a distribution of strengths. As such, indirect transitions are also possible. These transitions involve strength fluctuations which shift the balance of the transition criterion to be more favourable.

In the ten possible phenotype transitions from Figure 3.12, only the heterotetramer → 12-mer step meaningfully displayed an indirect transition. As discussed in the main text, the average strength states for this transition leads to the misassembly of the 10-mer, hence the abysmal transition rate. However, transitions would still be possible if the averages were perturbed enough, such as:

$$\phi_{12}\left(\hat{S} \geq \hat{S}^{*}(T)\right) > F^{-1/\gamma}$$

Where $\hat{S}^{*}$ is the minimum perturbed strength state, which has T dependence. The probabilities of these fluctuations are directly related to the steady-state distribution $\underline{\pi}_{\mathrm{PF}}$, from the Markov chain analysis. The sum over the states greater than $\hat{S}^{*}$, i.e. $P\left(\hat{S} \geq \hat{S}^{*}\right)$, gives the total probability an individual has to successfully transition on average. An example is shown in Figure C.2, where the transition criterion is only satisfied if $\hat{S} > \hat{S}^{*} = 0.775$.

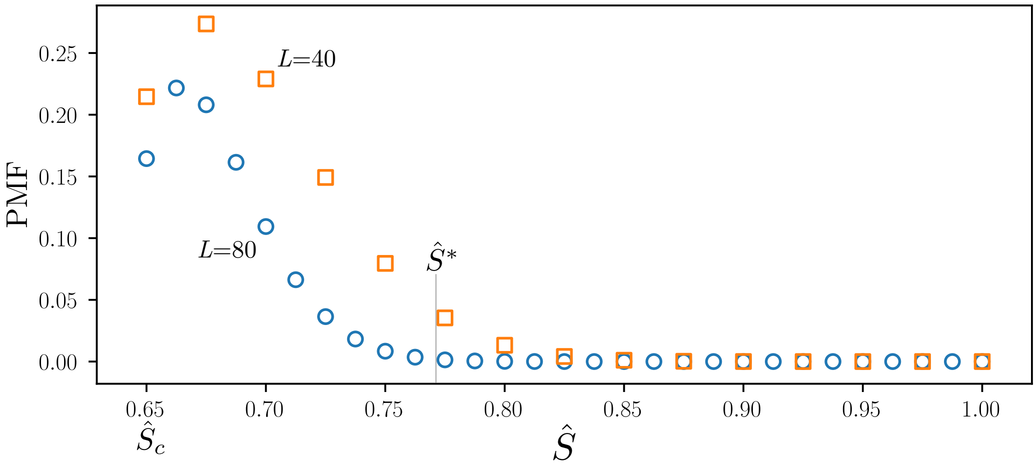


Figure C.2: The probability mass function (PMF) for the steady-state distribution of interaction strengths, $\underline{\pi}_{\mathrm{PF}}$, at $\hat{S}_c = 0.65$ for $L = 40$ and $L = 80$. For larger L, there are more states ($N = \left\lfloor L \cdot \left(1 - \hat{S}_c\right)\right\rfloor + 1$), and hence the probability of each state tends to be reduced. Fluctuations above $\hat{S}^{*}$ enable indirect transitions.

The probability quickly decays, especially for longer binding sites, and so indirect transitions are rare. In the case of the only observed indirect transition, the probability of being in a sufficient strength state was approximately 5%. Some of these transitions still failed as the transition criterion was only lightly satisfied with a small fitness advantage. Hence the observed 3% success can be tied directly to the out-of-equilibrium strength states of some individuals.