The Evolvability of Gene Regulatory Networks for Cortical Arealisation

A Thesis Submitted

for the Degree of

DOCTOR OF PHILOSOPHY

in

The Department of Psychology

by

Daniel John Whiteley

The University Of Sheffield.

to

FACULTY OF SCIENCE

THE UNIVERSITY OF SHEFFIELD

December 2022

ACKNOWLEDGEMENT

I'd like to express my immense gratitude to my supervisors, Stuart Wilson and Hannes Saal, along with my lab mates Seb James, John Brooke and Alejandro Jiminez-Rodriguez. I couldn't ask for a better group of people to explore the mysteries of cortical development! They have supported me through every technical, mathematical and motivational problem I've thrown at them.

Thanks also to all the family and friends who have encouraged me with kind advice at every step of the way.

I am also grateful to the Department of Psychology, Sheffield, for allowing me to do this project and providing all the required facilities. Along with the Collaborative Activity Award, Cortical Plasticity Within and Across Lifetimes, from the James S McDonnell Foundation (grant 220020516).

Dan Whiteley

December 2022

ABSTRACT

Computational models of gene regulatory networks are used to explore the interactions of development and evolvability. Pattern formation in the neocortex is modelled using Boolean Networks, Artificial Neural Networks and a novel Artificial Gene Network. The computational similarities between real gene networks and these models are described. This raises potential applications for reverse-engineering pattern forming gene networks, illuminating their unknown structure through statistical analyses. The ability of these networks to evolve is investigated under various conditions. The structure of the genotype to phenotype map in these artificial cases is probed to attempt to answer the question of what it is about this developmental system which makes it so highly evolvable?

Contents

Chapter 1

Introduction

Consider that your body contains thirty trillion cells, each of which was built for a particular function depending on where in your body it formed. Also consider that every one of those cells contains the same DNA as the original fertilised egg cell from which they came. What this means is that every cell is built using the same DNA 'instruction manual', however where that cell is positioned in the body determines how the 'manual' is read. A large part of understanding biological development is therefore understanding how DNA can be read differently in different cells, causing the expression of different genes. One of the most remarkable things about development is how exact it is in creating an organism from a single cell. So much so that it may not be easy to spot the differences between two identical twins. Development is therefore a highly accurate computational process, which takes the code of DNA and outputs a beautifully complex pattern of proteins, which is the organism. Understanding part of this process in detail, and therefore being able to model it accurately, is the aim of this work. From there we are able to investigate

how mutations to the genome can effect the organism, and thus how developmental and evolutionary processes interact.

My hypothesis is that non-linear dynamical systems structure the genotype-phenotype map in a way that makes the system more evolvable. This raises the key questions of my research; what is the structure of the genotype-phenotype map due to development? What structure is required for evolvability? Do different modes of mutation provide differing levels of evolvability?

The approach of understanding the evolutionary potential of a system by uncovering the underlying dynamical processes has been proposed by Jaeger and Monk (2014). This approach has also been applied to specific systems. For example, by analysing the dynamical properties of the gap gene regulatory network in the fruit fly, the network is shown to be functionally modular, which likely improves evolvability (Verd et al., 2019). Knabe et al. (2006) uses biological clocks as his vehicle to investigate the evolvability of gene regulatory networks. Along these lines I would like to model a developmental system so that both its dynamical properties and evolvability can be investigated together.

For this purpose I need a case in development which a) has interesting evolutionary properties, b) has a fairly well understood developmental mechanism, and c) is computationally abstractable. I will demonstrate that cortical arealisation is an excellent case study for these purposes. We will investigate how gene regulatory networks are responsible for dividing up the cortex into functional areas while also being a highly evolvable dynamical system.

This introduction will firstly cover the biological mechanisms that take a sequence of DNA and produce a pattern of gene expression, a key process in development. This

will be explored in detail with reference to how the cortex in mammalian brains is divided up into different functional areas. Each region of the brain can be represented as a different combination of expressed genes. Furthering a survey of relevant ideas, I will discuss the evolutionary significance of such mechanisms of development. Finally, some background of the existing models of this phenomena will be explained, and a plan for this thesis will be outlined

1.1 From genotype to phenotype

Biological development is the process that takes the genotype and builds the phenotype. At the lowest level we start with a string of molecules known as DNA (Deoxyribonucleic Acid), and our destination will be a spatial pattern of proteins which form the different cell types that allow us to function.

One remarkable feature of DNA is that it encodes a signal for creating different proteins, and at the same time provides the chemical machinery which regulates the use of that code. In computing terms it is simultaneously a data input file, a program and part of the hardware that implements that program. Each of the two strands in DNA's double-helix is a sequence of the nucleotide bases, Adenine, Cytosine, Guanine and Thymine (A, C, T and G). Each base is the complement of a base on the other strand – A with T, C with G – which provides the structural bonding for the double-helix while also allowing a mechanism for the chain to be unzipped and duplicated, so that cell replication can occur. What is less well known is that these bases, A, C, T, G, perform two different functions depending on where in the DNA they are. In 'coding' parts of the DNA they code for the creation of proteins, as

each combination of three bases defines an amino-acid to be added to a chain. In 'non-coding' parts of the DNA, depending on their order, the bases create binding sites for different proteins to attach to. When a particular combination of proteins is correctly bound to the DNA the coding part can become active, thus regulating the creation of proteins. A sequence of DNA responsible for building a single protein, both the coding and non-coding parts together, is called a gene. The activation of the code 'reading' the DNA to create the protein is called transcription. The proteins which bind to the DNA to activate transcription are called transcription factors.

For our purposes it is sufficient to know that once a gene is activated, transcription begins and various biological machinery copy the coding sequence, transport it out of the nucleus and use it to build many copies of an individual protein. However, the details of the non-coding aspect of the gene are much more important to us. That determines how the gene is activated, and therefore the regulatory processes that determine which proteins are built where.

The non-coding sections of DNA account for 98% of a human's genome. Some of it has no known function and is not associated with other genes (junk DNA). Other non-coding sections are situated next to the coding sections which they regulate, these are called cis-regulatory modules (CRMs). There may be several CRMs next to each other which independently regulate the transcription of their coding-DNA. Each module can also be called an enhancer, and the region of DNA containing the modules is sometimes known as the promoter. At the beginning of the coding region is a region of around 100 base pairs which the RNA polymerase II holoenzyme can bind to. This is the key enzyme in eukaryotes which transcribes the coding DNA (Wray et al., 2003). The gene is by default 'off'. The task of the enhancers is to assemble various proteins together which help this enzyme be positioned correctly and bind. The

process of getting transcription started by binding the enzyme includes an incredible assortment of chemical reactions and physical contortions of the DNA strand, with proteins binding to each other and forming superstructures that hang off the DNA. This is why we are unable to tell how the regulatory process will unfold by simply looking at the sequence of DNA, unlike in the coding-section where the genetic code (for amino-acids) is easily readable. We can look at the sequence of DNA to identify binding sites for certain transcription factors, but how the combination of hydrogen bonds, protein-protein interactions, and DNA-looping will lead to transcription is unpredictable.

It is instructive to have a look in detail at the CRM for a protein in the fruit-fly, whose pattern of expression on in the developing larva marks out the boundary of a certain body segment. The gene is called eve and the sequence of DNA shown is for the second module that regulates its transcription. See Figure 1.1

The long sequence of bases in a cis-regulatory module provide binding sites for many transcription factors. In the case of 'eve' we see binding sites for 'kruppel', 'giant', and 'hunchback' proteins. Each protein can have multiple sites, each with a different role in regulating transcription, and each with a slightly different configuration of bases which effects the binding strength of the site. The initiation of transcription will ultimately depend on the correct combination of binding sites being filled, with some needing to be empty. In this way the gene performs a computation, the input being a certain combination of proteins bound to the DNA. The output is initiation of gene transcription.

A simple example of binding site interaction is when two sites overlap, the site on the left (for example) needs to be bound for transcription to begin, but the

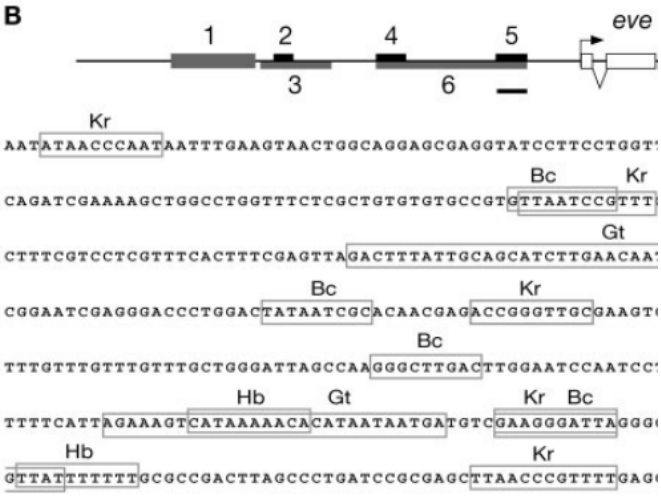


Figure 1.1: Binding sites for transcription factors Kruppel, Giant and Bicoid overlapping in the 2nd cis-regulatory module of the eve gene. The bases in between the binding sites determine their spacing, which effects their mutual interactions. Clearly simply inserting, deleting or substituting individual bases can have an enormous range of effects on the whole. Figure taken from (Wray et al., 2003), their Figure 3B.

presence of the protein on the right gets in the way. Another case of interaction is where sites that are nearby to each other are more likely to bind than if they were further apart, due to hydrogen bonding between the proteins. In this situation the number of bases between sites is important as it provides a spacing which effects the likelihood of binding. Using this mechanism of site spacing alone it has been shown that the module can represent any Boolean Logic function (Buchler et al., 2003), meaning transcription can be dependent on any complex combination of proteins being bound. However, we know that this is only the simplest interaction. There are other properties such as looping and contortion of the DNA strand when certain sites are bound, or bound proteins joining together with allosteric binding which then form larger structures. So it is seems plausible that any Boolean logic function

is likely possible to compute with this system.

By definition each module can initiate transcription independently. For example, in different segments of the fly larva the eve gene will be expressed, forming a stripe pattern with each module responsible for a different stripe. This modularity adds further to the potential computational complexity of a single gene.

Not only is the gene computing some Boolean function based on the discrete presence or absence of transcription factors, but the chemical process of binding allows for different concentration levels to be distinguished. Depending on the binding strength at the site, the probability of binding suddenly jumps up as the concentration of protein passes a critical threshold. As each protein can actually be bound by several different site configurations, each with a different binding strength, this allows the gene to fine-tune a kind of threshold operation in addition to its Boolean logic.

So each cis-regulatory module can be thought of as a computational unit. In other words each takes an input signal – the concentration levels of various transcription factors in the cell – and produces an output, the expression of a single gene. But the real power of this computational system becomes evident when we consider that those transcription factors are themselves the output of other genes. What this amounts to is a complex network of gene interactions, with each determining the activation of others. This is called the Gene Regulatory Network (GRN) and it is the key to understanding many phenomena in development.

To briefly summarise, each gene is responsible for building a protein, but by default the gene does nothing. In order to begin transcription it requires an exact combination of other proteins to be present, and these proteins will be present when

other genes have been expressed. Once this complex set of interactions between several genes settles down a pattern of gene expression emerges. The expression pattern might be stable, or it could change in time, going round in a cycle. The eventual pattern of expression thus determines various cell fates depending on the position of the cell in the body. In order to do this the system needs information relating to each cells' position. We will consider next how this is achieved.

If every cell contains the same DNA, and therefore computes the same gene outputs from the same input, how does a differentiated pattern of expression emerge across the organism? The answer is that the initial input also varies spatially. In the case of the fly larva, the system begins with a single protein (bicoid) being expressed in a high to low gradient due to the maternal environment (Driever and Nüsslein-Volhard, 1988). This simple differentiation across space sets each cell's gene regulatory network off along a different path, even though the DNA which encodes it is the same. The spatial differentiation unfolds with increasing complexity, starting with the simple gradient bicoid, then the 'gap genes', kruppel, giant and hunchback, which form smooth bumps of expression in different locations. Finally, dependent on the gap genes, the eve stripes develop. These allow each segment of the larva to be clearly demarcated and develop completely differently.

Let us now consider two theories which could explain how such stripes develop in the larvae, allowing the fly's body plan to develop. The first is due to Alan Turing and is known as a reaction-diffusion model (Turing, 1952). This process is thought to explain how complex natural patterns like those on animal skins are formed, for example stripes or spots. The model shows that complex stable structures can emerge from an almost homogeneous chemical primordium. A small perturbation will propogate and develop as the chemicals diffuse and react, until a dynamic equilibrium

is reached. The process is somewhat similar to the formation of standing waves. On the surface this seems like an elegant way for a system to establish complex patterns, only passive chemical reactions are required rather than gene regulation. However, at least in the context of cortical arealisation, a few problems with this model arise. Early on in embryonic development the tissue is small enough to allow diffusion, which can create an initial gradient. But the interactions needed to create cortical boundaries happen later, when the tissue is too large for proteins to diffuse across. Related to this is the issue that some proteins involved in marking out cortical areas stay confined to their cells, with very little diffusion. Another feature of Turing patterns is that the characteristics of a pattern would remain the same across organisms, like all leopards having spots. But the exact positioning of individual spots would not be the same for each organism as they are determined by small irregularities in the initial conditions. For these reasons the Turing reaction-diffusion model does not explain the types of pattern formation concerning us here.

The second theory is that cell fate can be determined from the 'positional information' in a graded expression profile (Wolpert, 1969, 1996). It is known as the 'French Flag Model' or Wolpert's positional information theory. The model begins the same way as with Turing, with an initial morphogen diffusing across the tissue early in development. This forms a gradient of protein concentration. There then needs to be a mechanism for each cell to 'sense' that protein concentration and therefore have information about its position in the gradient. In this way the gradient forms a coordinate system that cells locate themselves on. At certain thresholds of protein concentration, and thus at subsequent positions, the cells develop differently. Going from left to right one could imagine three cell fates developing at different thresholds, forming blue, white and red cells like a french flag.

At first sight the stripes of the fly larva should be a perfect example of Turing's theory, it is a much neater explanation of a repeating pattern than the fact that each stripe is formed independently through a thresholding operation on the gradient. But surprisingly experimenters have demonstrated the ability to express each module of the eve promoter by itself, alongside much experimental data on the precise gene interactions involved.

Therefore Wolpert's positional information theory is a keystone in our understanding of how complex morphological patterns emerge in the cases where they are initiated by a gradient. The theory is further supported by the explanation of how GRNs can use the initial gradient to compute clear boundaries thanks to their boolean logic and thresholding operations. However, Gordon and Gordon (2016) lay out seven critical points of weakness with the model. these center around how the gradient is initially created and then interpreted by cells in a sharp boundaried manner. We will now turn to how this is achieved.

Rogers and Schier (2011) present an excellent review looking at how these morphogen gradients are established and then interpreted. They explain that there are many possible ways a gradient can be established. The most simple is the Synthesis-Diffusion-Clearance model, whereby a morphogen emitted at one pole of a field diffuses across and is cleared either by a sink at the other pole or removed throughout the field by enzymes or chelators (Crick, 1970). However in the case of Bicoid, the gradient is due to mRNA transport, then local translation, not passive protein diffusion. The lack of diffusion is in fact what allows a robust boundary to form (Lipshitz, 2009). Other modes of gradient formation are possible. For example, there is an interesting case where Chordin is expressed at the dorsal end of the frog embryo, where it forms a complex with BMP and 'shuttles' the morphogen to the ventral end, form-

ing a BMP gradient where it was uniformly spread before (Lewis, 2008). Finally, it seems feasible that while some gradients, once formed through diffusion, could initiate other gradients through intracellular dynamics such as gene regulatory networks. Ashe (2006) explains the mechanisms by which genes respond to the gradients at a particular threshold. This is how the cell interprets its position in the French Flag model.

While Wolpert's Positional Information is clearly not the mechanism responsible for all developmental processes, it is hard to argue it is not involved in the formation of the cortical map. There may be unsolved questions as to how transcription factors develop their gradients, but it cannot be said that these gradients do not exist. As for the thresholding interpretation of the gradient by individual cells, I hope to convince you that gene regulatory networks have the capability to do just that.

1.2 The development of the cortical map

The fruit fly provides a simple example of positional information at work in an essentially one dimensional spatial domain. The differentiation of cell types only occurs on the anterior-posterior axis. I'd like to focus on a two dimensional case which is of crucial relevance to mammalian evolution, neocortical arealisation. All species of mammals have the outer sheet of their brains divided up into different areas depending on what informational processing is important for their survival. For example, primates have proportionately larger visual areas, bats have larger auditory areas, and mice have larger somatosensory areas. According to the protomap hypothesis (Rakic, 2009), this cortical division is predetermined by a 2D expression pattern of

genes early on in development. If this is true then we have a perfect system to model the whole process from genotype to phenotype. Not only is the output a map of gene expression, but it also seems that the process for obtaining the map involves thresholding operations on gradients as in the case of the fly.

Neocortex has a distinctive six layer structure that is present only in mammals (Lui et al., 2011). It is responsible for higher order brain function. It is particularly important for humans as it has expanded enormously, taking up three quarters of the human brain, hence why the cortical sheet is so folded to fit in our skulls, whereas in other mammals it is smooth. The neocortex is the most important part of the brain in terms of the cognitive advances acheived by humans. The neocortex has a role to play in all of these very human qualities: syntactical-grammatical language, symbolic thought, self-reflection, long-term planning, autobiographical memory, theory of mind, and art (Call and Tomasello, 2008).

The neocortex is not the same across its surface. It has regions which are specialised for processing different kinds of information. In 1909 Brodmann described 43 different areas in the human brain by looking at cell stains and finding visible differences between regions (Zilles and Amunts, 2010). Each region will have different input and output connections and different neuron types. Different species have different sizes of areas depending on their specialities, and also different numbers of sub-fields within each area. The number of neocortical areas in mammals varies between 15 and 200 according to the definition (Krubitzer and Prescott, 2018). Areas important for language processing in humans can't be identified in rodents (Rakic, 2009), but interestingly they are present in chimpanzees (Sousa et al., 2017). While there are key differences, the general cortical plan is homologous across mammals, having the same primary areas and approximate positioning. This has been shown

by Krubitzer and Kaas (2005) through comparative studies of extant species.

While there is evidence of extrinsic factors in cortical arealisation, the areas are coarsely determined by the genome. The mechanisms through which the genes cause the eventual cortical map strongly constrain what evolution has to work with. This can be seen in the homology across mammals, and also in special cases such as the blind mole rat which still retains its primary visual area (Bronchti et al., 2002). One reason for these constraints is the multiple use of developmental genes, known as 'pleitropy'. If one gene plays multiple roles in development any variation in its expression may have fatal side effects.

To introduce how the cortical fields develop, we begin with a sheet of neuroepithelial cells located in the ventricular zone of the dorsal telencephalon. The cells in this sheet will divide and diversify, expanding outwards radially to form the six layers of neocortex. A key finding was that the progeny of these initial cells migrate via a scaffolding of Radial Glial Cells, with the closest layer (layer VI) being formed first, and so there is little tangential or horizontal movement. This is known as the radial unit hypothesis (Rakic, 1988). The first neurons to be formed are the subplate neurons around E11.5 in mice (Greig et al., 2013), so an important question is whether the areal specification of the cortex is already determined by this time. There is indeed evidence of differential gene expression across the telencephalic field which could provide such a 'protomap' by this early stage (Sansom et al., 2005).

Another possibility, known as the 'protocortex' or 'tabula rasa' hypothesis, is that the neurons arising radially from the neuroepithelial sheet do not yet have their areal identity determined, and it is extrinsic factors such as thalamic innervation or sensory cues which will divide the cortex into its boundaried regions (O'Leary,

1989). Comprehensive reviews of this cortical development process and also how the protomap may arise from patterns of gene expression are Sansom and Livesey (2009); Mallamaci and Stoykova (2006); Sur and Rubenstein (2005); Rakic (2009); O'Leary and Nakagawa (2002). The question of whether cortical arealisation is determined early on by gene expression patterns (protomap), or whether it is an effect of later mechanisms such as sensory input (protocortex), is crucial to our study of the role of the genome in cortical development. So, what is the evidence for each proposition?

The protocortex hypothesis was initially suggested by Van Der Loos and Woolsey (1973). They damaged individual whiskers of mouse pups at birth, and found the corresponding cortical areas (barrels) to be absent two weeks later. This shows that sensory input can have an effect on the parcellation of cortex.

The theory was further developed by O'Leary (1989), who pointed to important experiments whereby rerouting retinal information to somatosensory thalamus (Métin and Frost, 1989), or to auditory thalamus (Sur et al., 1988), causes S1 or A1 to respond to stimuli in ways normally seen in V1. This could suggest that primary cortical areas process information from the thalamus in the same way, or alternatively that they would be different except that visual input has altered their development. But either of these explanations supports the theory that the cells in these developing regions of neocortex are to some extent pluripotent (O'Leary, 1989). Further evidence supporting this theory is that cells from mouse embryonal visual cortex were transplanted into the somatosensory area, where they went on to aquire barrel features (Schlaggar and O'Leary, 1991).

The most persuasive evidence for the protomap hyposthesis is that mice with no thalamocortical projections, as a result of gene knockouts, still retain normal cortical

arealisation (Nakagawa et al., 1999). Also, taking explants from different regions of early developing cortex and then growing in vitro shows that they are committed to expressing the molecular markers for their region (Arimatsu et al., 1992). More recently Pattabiraman et al. (2014) has found a set of enhancer elements active by E11.5 which drive gene expression in particular protodomains. There is now substantial evidence in favor of the protomap hypothesis. Although there are not specific transcription factors whose expression form the boundaries of this map, the expression of the transcription factors is graded and enhancers bring together this positional information to determine cell fate. There remains a role for the protcortex hypothesis in fine-tuning the cortical map and determining structures such as whisker barrels internal to the region, but for our purposes we can constrain the investigation to the protomap formation in the neuroepithelial sheet.

Cortical maps are broadly determined by the genome, consistent with expectations from the Positional Information theory of Wolpert. This being so, what are the key gene's involved and how are the initial gradients set up and interpreted by gene regulatory networks? In the cortical primordium one of the key genes found to provide positional information is FGF8, which is a ligand that is secreted from the commissural plate prior to E10 (Mallamaci and Stoykova, 2006). Bachler and Neubüser (2001) predicted its role in determining the rostral to caudal fate of cortical regions. An important experiment which demonstrated this is that when FGF8 was artificially expressed caudally (whereas it is naturally expressed rostrally), it caused a mirror duplication of cortical areas, including the entire internal structure of the somatosensory region (Fukuchi-Shimogori and Grove, 2001). This aligns with the positional information model, if we assume the gradient of Fgf8 from high rostral to low caudal is imparting a coordinate system, then mirroring that gradient should

lead to a mirrored cortex, which it indeed did.

Rakic (2009) categorises the genes involved in patterning the cortex into 3 levels. There are the 'Patterning Centres' such as Fgf8, Wnt and BMP which are secreted from the cortical hem or anti-hem, or anterior neural ridge. There are the 'Progenitor Gradients', such as Coup-tf1, Emx2, Pax6 and Sp8. And finally the 'Cortex Markers' which form more discrete boundaried shapes, such as ROR-beta. The patterns that the products of these genes form can change over time and be different in different layers of cortex, but there is clear evidence of the kinds of gradients required for the positional information model, and evidence of complex networks amongst these genes.

Starting with the patterning centers from E9.5 in mice, Fgf8 and 17 are secreted from the rostromedial border, Wnts and BMPs caudomedially, and laterally Sfrp2 is secreted. Of these only Fgf8 is shown to direct arealisation (Greig et al., 2013). Fgf8 diffuses from its anterior source, creating a gradient in the mouse neocortical primordium (Toyoda et al., 2010). Although the human brain is eventually much larger, making gradient formation by diffusion more difficult, at the equivalent developmental stage the neocortex is only 0.5mm long. The Fukuchi-Shimogori and Grove (2001) experiment, mentioned above, very directly showed how FGF8 can impose a coordinate system on the developing cortex. So much so that the addition of FGF8 at the opposite pole causes a mirror duplication of the entire cortex.

Of the 'Progenitor Gradients' there are four in particular which have been shown to alter position and size of cortical areas. They form well defined gradients going in opposite and perpendicular directions, ideal for a coordinate system. Firstly lets look at Emx2 and Pax6, which form one opposing pair, Emx2 has high expression

mediocaudally which decreases in a lateral rostral direction, while Pax6 shows the exact opposite profile. When a mouse has a mutation to remove Emx2 its rostral areas expand and caudal areas contract along the direction of the gradient. Again, Pax6 does the exact opposite (Bishop et al., 2000).

Similarly Sp8 and Coup-tf1 also form an opposing pair, almost perpendicular to the Emx2-Pax6 pair. Sp8 has high expression in the mediorostrally which decreases in a lateral caudal direction, and Coup-tf1 does the opposite. By expressing these transcription factors where they are not usually expressed it has been shown that Sp8 and Coup-tf1 reciprically inhibit each other (Borello et al., 2014). Again knockout experiments show shifts in the cortical map in the expected direction for both Sp8 and Coup-tf1 (Sahara et al., 2007; Armentano et al., 2007). For the main experimental results implicating Emx2, Pax6, Coup-tf1 and Fgf8 in mice see O'Leary and Nakagawa (2002). Later on in development there are combinations of gene products which mark out well-defined cortical areas. Suzuki et al. (1997) found that each cortical area, and also sub-cortical regions, were delineated by a specific combination of Cadherin 6,8 and 11 expression in the postnatal mouse. However there is no such explicit demarcation at the E11.5 stage, the time by which we require the protomap to be formed. Rather the progenitor cells work with gradients for positional information (Sansom et al., 2005). Transcription factors are regionally enriched rather than area specific at this early stage (Mallamaci and Stoykova, 2006). However, patterns of enhancers, i.e, the cis-regulatory elements that control expression, do match the eventual cortical fields (Pattabiraman et al., 2014). So the experimental evidence seems to support the protomap theory.

We can now describe how organisms develop the cortical map phenotype from the genotype. Starting with a sequence of DNA in every cell, the cis-regulatory modules

activate their genes according to their position in a gradient. This determines the expression pattern of a class of genes such as the Cadherins which mark out clear boundaries for the cortical areas to form differentially. Understanding this process will allow the creation of an accurate model of the developmental process, it will be possible to alter the genotype and observe changes to the phenotype. With such a developmental model we will be able to run evolutionary simulations, investigating metrics of evolvability and phenotypic change. From such experiments we can make inferences about the interactions between evolution and development. To see the potential in these investigations I will now turn to the evolutionary aspect of this study.

1.3 Evolutionary Development

In evolutionary theory variation is key. The types of variation produced in a population are strongly linked to the populations' ability to evolve. Evolvability is defined as "the capacity of a population to generate adaptive heritable genotypic and phenotypic variation" (Nehaniv, 2005). Originally, Darwin's theory said nothing of how variation arises in a population. It seemed sufficient that it clearly exists in nature. Evolutionary theory progressed in the 20th century to include genetics and inheritance, resulting in the currently accepted paradigm known as the Modern Synthesis. Within this current theoretical framework the origin of variation is still underplayed, it is assumed that there is a simple mapping between genotype and phenotype which means that small genetic changes will produce a neat bell-curve of phenotypes (Salazar-Ciudad, 2006). However, this implicit assumption is questionable. The process of development, which takes a genome and creates an organism,

is extremely non-linear and discontinuous, so a small mutation could be fatal and a large mutation could have zero effect. New ideas in evolutionary theory such as the Extended Synthesis (Pigliucci and Muller, 2010) and Evolutionary Development seek to explain the origin of beneficial variation, and therefore evolvability. The key to this explanation will be understanding development and how mutations lead to variations.

Evolution appears to be mainly operating at the level of GRNs, changing links in the network rather than changing the expressed proteins. Changes in the genome could effect 'coding sequences' of the DNA which change the proteins expressed, or 'non-coding sequences' which may effect the logic of the network of gene interactions. King and Wilson (1975) showed that between humans and chimpanzees the majority of sequence differences are of the non-coding type, see Nord et al. (2015) for review. In fact, going back hundreds of millions of years to the last common ancestor of bilaterians there has been a largely conserved toolkit of genes responsible for body patterning. The key evolutionary changes involve adapting the interactions between genes, rather than the proteins expressed (Carroll, 2008). Comparisons of modern human and Neanderthal genomes found changes that gave rise to only 87 proteins with amino-acid substitutions, and these proteins were found to be enriched in the ventricular zone during cortical development (Prüfer et al., 2014).

Gene regulatory networks thus offer an important level of abstraction to study the interplay of development and evolution. They are the fundamental substrate of evolutionary change, but we also know their importance for the development of cortical areas. Jaeger and Crombach (2012) wrote "Physiological and developmental processes define the evolvability of a trait: can it evolve at all? At what rate? Can it evolve in a gradual, continuous manner? To what extent? In which ways can it

change? And last but not least: can evolvability itself evolve? These questions define some of the central problems of evolutionary developmental biology". Potentially if we can successfully model these GRNs then we may be able to direct the model to address some of these important questions.

1.4 Models of Gene Regulatory Networks

In the overall process from genotype to phenotype there are several stages that need modelling, and there are differing levels of model abstraction depending on the purposes required. The core of any model must capture the dynamics of a single gene. That is, how gene transcription is initiated depending on transcription factor concentration. There are then additional assumptions required for modelling the network dynamics of multiple genes. Finally, if the input into this model assumes that some genes have a linear expression gradient then this assumption must also be accounted for.

Models of single genes require mathematical elaboration of the chemical kinetics underpinning gene transcription, whereas considering multiple genes involves using some form of logic processing. Both of which can be adapted to get the desired level of biological accuracy. For a thorough examination of these models see (Bolouri, 2008).

We can split the mathematics of gene transcription into two parts, firstly considering the probability of a single transcription factor binding to the DNA, and secondly the effect of multiple bound transcription factors on the initiation of transcription. The first part is fairly straightforward. In a steady state the associa-

tion of the transcription factor with DNA happens at the same rate as dissociation, both of which are proportional to the concentration of those molecules (the Law of Mass Action). Therefore concentration of bound TF-DNA, $[TF - DNA_{bound}] = K_A \times [TF] \times [DNA_{free}]$, where K_A is the equilibrium association constant. The number of interest to us is the fraction of DNA which is bound as this relates to the fraction which will be successfully transcribed, which can be found by manipulating the above equation to give:

$$P(binding) = \frac{[TF - DNA_{bound}]}{([TF - DNA_{bound}] + [DNA_{free}]} = \frac{K_A[TF]}{(K_A[TF] + 1)}. \tag{1.1}$$

If all it took to initiate gene transcription was a single bound transcription factor then this equation would give the probability of transcription based on the concentration of that transcription factor. However, we know that transcription actually involves many binding sites of different types. This affects the overall probability of transcription by requiring some consideration of the logical dependence of the binding sites with each other. But it also affects the probability of binding for each independent site, this requires adapting the equation above by including a 'cooperativity' factor known as the Hill coefficient, n:

$$P(binding) = \frac{K_A^n[TF]^n}{(K_A^n[TF]^n + 1)}. \tag{1.2}$$

The convention is to instead use the dissociation constant K_d which is useful as it amounts to the concentration level at which P=50%. This equation, known as the Hill-Langmuir equation (Stefan and Le Novère, 2013; Gesztelyi et al., 2012) is crucial to all the models in this work:

$$P(binding) = \frac{[TF]^n}{(K_d^n + [TF]^n)}. \tag{1.3}$$

The first model we shall mention then is the application of this equation to understand the temporal dynamics of transcription. By equating the probability of binding with a change in the rate of protein produced we can obtain an Ordinary Differential Equation (ODE). Considering the causes and effects of multiple elements in the gene network (proteins, DNA, mRNA), we can build a whole system of connected ODEs which can be solved to give the time profiles of each chemical concentration. However, for the purposes of spatial patterning we are not too concerned about the time variance of the expression profiles. We are modelling systems which converge over several hours into a final stable pattern, unlike other GRNs which use their cyclic temporal patterns for different functions.

The second model to consider, the Boolean Network (Kauffman, 1993), stems from making a basic simplification to the above Hill function. Figure 1.2 shows the probability of binding (which we can associate with the concentration of protein, P) depending on the concentration of the transcription factor, A. The green dotted line shows the case of a singe basic independent binding site, the blue line includes cooperativity, and the red line makes the simplification that transcription can be considered binary, on or off.

Such an assumption makes it possible to combine the effects of genes in a network or circuit of logical Boolean functions. The pattern of activation that the genes create in such a network is an emergent phenomena which closely resembles the activity of real cells. Even networks which are randomly assembled with random logical functions have interesting dynamics which mimic real biological behaviour

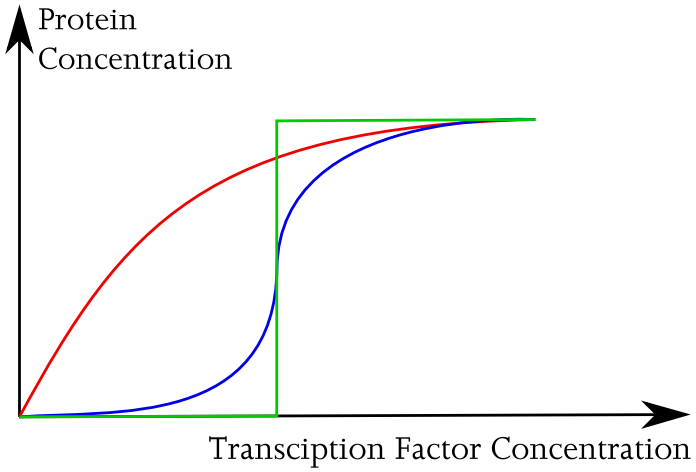


Figure 1.2: The concentration of the protein produced depends on the concentration of transcription factor. This figure shows the dependency under three different assumptions. Single independent binding site (red), multiple cooperative sites (blue), binary (green).

(Kauffman, 1969). Again, this can be useful for modelling temporal dynamics, but in this case there are also examples of investigating spatial patterns, even cortical development itself.

To see how Boolean Networks function consider a set of three genes, each could be on (1) or off (0). Whether each gene is active or not depends on the activity of all the others in the timestep before. Perhaps gene A will turn on when gene B and C are both on, we can write out such a relationship as the boolean logic function A = B AND C. Or we can write out the full truth table as in Figure 1.3.

If we combine the truth tables for all three genes then we get a state transition table, meaning that when the input state is given by the three columns on the left, the output state is given by the three on the right (Figure 1.4).

Note that in this case I've filled the output columns for B and C randomly.

Input (time = 0)			Output (time = 1)
Gene A	Gene B	Gene C	Gene A
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Figure 1.3:

Input			Output		
A_t	B_t	C_t	A_{t+1}	B_{t+1}	C_{t+1}
0	0	0	0	1	0
0	0	1	0	0	1
0	1	0	0	0	1
0	1	1	1	0	1
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	0	1	0
1	1	1	1	0	1

Figure 1.4:

When the output state is the same as the input state the system stops, for example if we start with 000, we can follow through the transition table – 000 → 010 → 001 → 001. The system reaches a stable point 001. There are other starting points which also lead to 001, for example 110 → 010 → 001 → 001. We say that 001 is a point attractor and that all the states that lead to it form the basin of attraction. Not all attractors are single points. If we start at 111 we get 111 → 101 → 011 → 101 → 011 → 101 and so on going back and forth forever, such an attractor is called a limit cycle or orbit. It is possible to visualise the whole state space with all the transitions, which makes it easy to see all the basins of attraction and attractors at once (Figure 1.5).

The appeal of Boolean Nets in their early days was how a system with countless possible states (in the case of many genes) self organises into a small number of attractor states, much like real biological cells. Later research developed even tighter links with biology. In the case of Drosophila melanogaster (fruit fly) it was possible to build a Boolean Network from the known interactions between genes and accurately predict the resultant expression patterns, showing the crucial role that particular genes had on the system (Albert and Othmer, 2003). As well as giving insights into individual genes and interactions the model allows the study of global dynamics, such as how the Yeast cell cycle is robust to perturbations thanks to the large basin of attraction leading to a point attractor (Li et al., 2004). But the example of most relevance to this work is the effort to determine the interactions responsible for initialising cortical arealisation (Giacomantonio and Goodhill, 2010). As we have seen, the development of cortical areas relies on a set of genes forming linear gradients (Coup-tf1, Emx2, Pax6, Sp8). If the gradients are simplified into two-domain systems with values of 1 or 0, it is possible to find all candidate networks which produce the

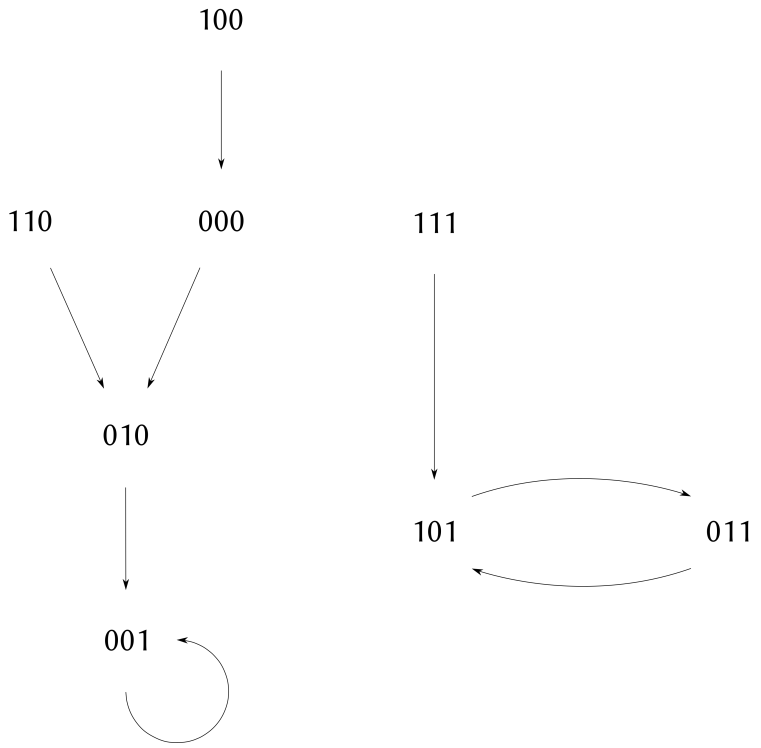


Figure 1.5: Visualisation of state space with two basins of attraction, the left leading to a point attractor, the right leading to a limit cycle. Note that this is a representation of the state transition table in Figure 1.4.

desired patterning. Giacomantonio and Goodhill (2010) used this approach to show which interactions are most prevalent in successful networks.

Boolean Networks are the most highly abstracted models used to research gene regulation, but they capture something fundamental about the global dynamics of such systems. They also give a basic framework for how a phenotype can be generated from a genotype and can therefore be useful in studying evolutionary dynamics as well.

The final model to be reviewed is useful because it involves a network of nodes

whose output is based on whether its inputs pass some threshold, just like the requirement for Wolpert's Positional Information, and also like how the Hill function above shows how real genes are activated. The model was initially invented to mimic the behaviour of neurons in the brain rather than genes, however the underlying mathematics is applicable to both. Neurons have the property that they only emit their electrical signal if the sum of all the incoming signals passes some threshold. When such neuronal units are combined in a network they have immense computational power which forms the basis of most AI-driven technology today.

Artificial Neural Networks are a set of nodes connected by links of differing strength. The activity or value associated with each node is calculated by adding together the values of its input nodes multiplied by their connection strengths, or 'weights'. The value is then adjusted by a non-linear squashing function $\sigma()$, to keep the output in a certain range.

The network can be structured hierarchically, with layers of nodes feeding in to each other, or it could be recurrently connected in a cyclical nature. Either way there must be some nodes that are designated input nodes and some must be output nodes. The purpose of the network is to compute the correct output given a particular input. This is achieved when the weight values are gradually improved and the network is 'trained' to produce the correct output. Apart from the similarity of the thresholding operation with that of a gene, this model is related to pattern formation because the input nodes can be fed with a coordinate position, and the output nodes can represent the gene expression at that position. This allows us to train the network to produce a certain image, for example a cortical map, using similar computations to the genes that do it for real.

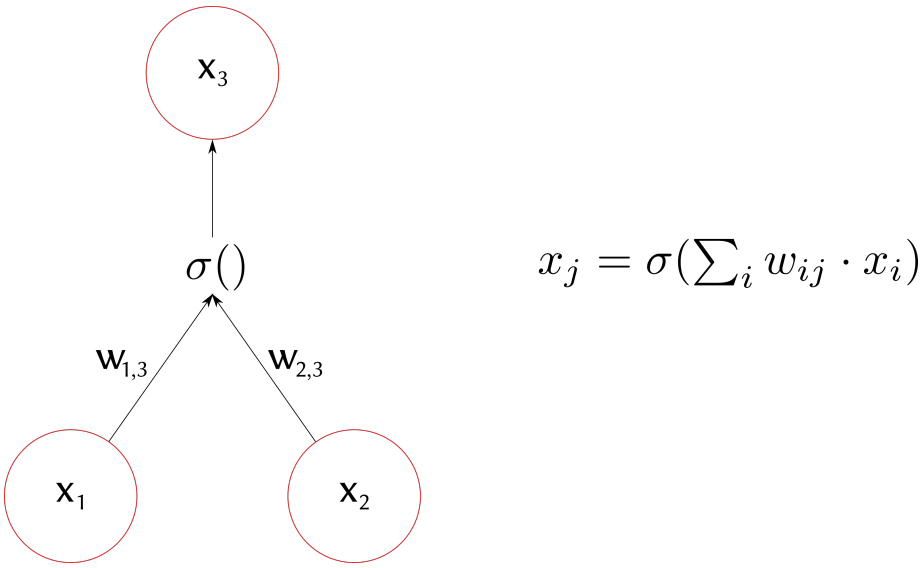


Figure 1.6: A single perceptron, the unit of a neural network. The output of a node is the squashed weighted sum of its inputs. The squashing function that is often used, $\sigma()$, is the logistic or sigmoid function. (Rosenblatt, 1958; Rumelhart et al., 1986)

There has been a lot of work using Artificial Neural Networks as an AI tool for infering gene interactions from statistical data, but there has been little on making an equivalence between nodes and genes creating spatial patterns. The usual purpose of neural networks is to capture the statistics of a set of data. The dataset would be divided into two parts, a training set of many examples on which the network learns the correct set of weights, and a test set on which the success of those weights at predicting the output can be measured. What is important is the value of the output nodes, the intermediate nodes form a kind of 'black box', often of little interest. However, when modelling gene nets for spatial patterning they are crucial. They represent the genes which position the cortical areas. Just like Giacomantonio and Goodhill (2010) did for boolean nets, we can generate hundreds of networks which produce the same cortical map. From this ensemble we look for patterns in

the hidden nodes, or for information on how successful networks are depending on their structure and size.

In one model of a simplified CRM, in which transcription factors only effected each other through weak hydrogen bonds, it was shown that there is a direct mathematical equivalence between that system and a type of Neural Network called a Boltzmann Machine (Buchler et al., 2003). The paper shows that the simplified CRM, such as a bacterial rather than eukaryotic gene, is capable of computing any boolean function by adjusting the binding strength and cooperativity parameters only. In this case the equivalence to a neural network is found by making each node represent a binding site, the value of each node is related to the concentration of transcription factor, and the connection weights are related to the cooperativity between the two sites. The Boltzmann machine differs only slightly from the neural network described above. Instead of each node having a continuous value it has a probability of 'firing', delivering a discrete result of 0 or 1. While in this case the fruitful analogy is between individual nodes and binding sites of the gene, when modelling spatial patterning it also makes sense to have each node represent a whole gene, as it is the whole gene that computes the thresholding operation.

1.5 Upcoming investigations

We have now laid out the foundations for the upcoming research. Namely, that gene regulatory networks are responsible for cortical arealisation, and they do so as laid out by Wolpert's positional information theory. Cortical development is an appropriate system for us to work with since the phenotype can be mathematically

reduced to a set of areas on a 2D surface. Moreover, we have sketched some details of the models of GRNs which can be used to represent a bridge between the genome and this simplified phenotype.

With these foundations in place we can begin an investigation into the properties of dynamical systems that impact evolvability.

A good starting place is with the most simplified abstract model of GRNs, Boolean Networks. While we know the biology is much more complex than this model's binary expression patterns, they capture a fundamental property of dynamical systems, attractors. Attractors can give a greatly condensed set of outputs for a large space of initial conditions. So in Chapter 2 we ask the question of how evolvability may be aided by attractors structuring the genotype-phenotype map.

The next step is to improve the biological plausability of the model to see if these features can be found in more realistic systems. In Chapter 3 we develop a model based on neural networks which will allow us to make comparisons with continuous 2D phenotypes. Does such a model align with the complexity and structure of real GRNs for cortical arealisation?

Taking the new neural network model as a tool, we once more turn to evolvability by investigating the role of phenotypic distribution and mutation in evolution. Chapter 4 aims to shed light on why gene regulatory networks are so capable of producing adaptive variation for natural selection to work with.

Finally, we renew the model to reflect the real logical mechanisms underlying gene transcription as much as possible. Under this new framework can we learn more about why real GRNs are structured as they are? And how do differing modes of mutation effect evolvability in this case?

Chapter 2

Evolvability and self organisation in Boolean networks

This chapter is a reprint of the paper, *Limit cycle dynamics can guide the evolution of gene regulatory networks towards point attractors* (Wilson, James, Whiteley, and Krubitzer, 2019). Originally published in Nature Scientific Reports. A new paragraph has been added to the Discussion, beginning "A criticism of this model...".

It follows Giacomantonio and Goodhill (2010) in simplifying the cortex into two domains, anterior and posterior, within which five genes have binary expression. The five genes interact in a Boolean network which is gradually altered by an evolutionary algorithm. The target of evolution (peak fitness) is a set of states which represent a point attractor. However the characteristics of the developmental process allow for limit cycles of states which approximate the target and thus make the system more evolvable.

This work is an extension of my MSc Thesis, as well as being co-authored by two other people in my lab. Therefore I should be as transparent as possible about the novel contributions I have made for this chapter, and those that others have made. The work for my masters took the framework for investigating cortical arealisation that Giacomantonio and Goodhill (2010) set out, but developed a new methodology. The first change was to use a synchronous deterministic model, and secondly to probe the phenotypic structure using evolutionary algorithms. This gave three results: that the set of successful networks is smaller but contained within those of the Goodhill model, that the fitness landscape has a structure conducive to evolution, and that the evolutionary algorithm finds peak fitness 39 times faster than random search.

The current work took those results as a foundation and investigated further. Firstly, multiple different fitness functions were trialled, we settled on the one we felt to be most biologically plausible. Secondly, the sensitivity of the model for different target states and asynchronous updating was tested. This is the core work that I did. In addition to this Seb James did further work on testing the complexity of the networks using the Quine-McCluskey algorithm, and tested the model for contexts > 2 (i.e. not just anterior/posterior cortex). I'm also grateful to Seb James for porting the code to C++ from Python and speeding it up immensely. Stuart Wilson drafted the majority of the text which the three of us edited collaboratively. In short, while this work is clearly a team effort I feel justified in including it here as I was responsible for the development of the methodology and its implementation.

2.1 Introduction

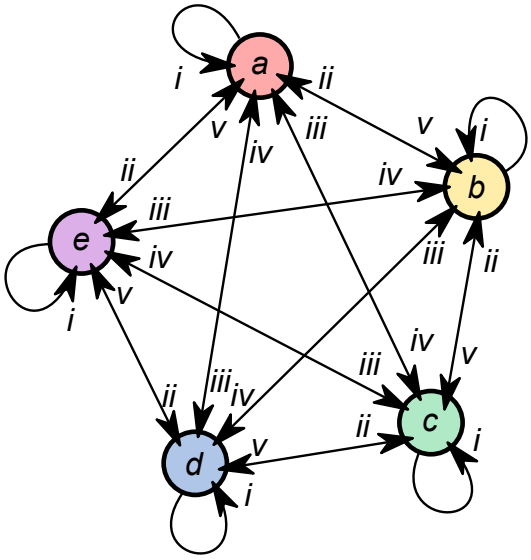
Self-organization and natural selection are fundamental forces that shape all biological systems. Self-organization describes a dynamic in a system whereby local interactions between components collectively yield a global order that is unobservable, in its entirety, to the individual components. Therefore, self-organization describes dynamics that occur across all spatial and temporal scales throughout the lifetimes of all organisms. Natural selection is instead a description of dynamics that occur primarily between lifetimes, via the communication of genetic information from organisms to their offspring. It is clear how natural selection can operate on the self-organizing processes by which organisms develop and compete, because information passed on by genetic inheritance specifies the interactions within and between those self-organizing processes. But the extent to which self-organization can operate on natural selection is not yet understood. Can selection, modelled as a global optimization of the genotype by fitness maximization *between lifetimes*, exploit the emergence of structure in the local mapping from genotype to phenotype that occurs *within lifetimes*? Here we show that fitness landscapes can be modified by the intrinsic properties of dynamical network self-organization, via a simple, biologically plausible mechanism that is compatible with conventional descriptions of evolution by natural selection.

Consider a network of n interacting genes and assume for simplicity that their expression levels may be either high or low only. The network interactions can be specified by assigning to each gene a truth table that determines its next expression level in response to each of the 2^n possible patterns of expression. The developmental dynamics of the network can thus be completely specified by a string comprising

$N = n2^n$ binary digits. Whilst acknowledging the obvious limitations of the analogy, we can, for convenience, refer to this binary string as a 'genome'. The genome has 2^N possible configurations. See Fig. 2.1.

The dynamics in these networks self-organize to reveal attractors (Kauffman, 1969, 1993; Gershenson, 2012). From a given initial state (a state is a pattern of n binary expression levels), the network activity will eventually settle, either into an endless repetition of a single state, known as a point attractor, or into a limit cycle, where a specific sequence of states repeats endlessly (see Shimojo et al. (2008); Imayoshi et al. (2013)). In the broadest terms, different initial states represent different contexts in which the network dynamics may develop, as determined by factors extrinsic to the network, such as the transient influence of another gene or gene network, differences between cell or tissue types, or different temperature or chemical conditions (food, oxygen, hormones etc.). Thus we might consider a mapping from a given initial state to a point attractor to constitute a robust response of the network in that context. Assuming that the initial states are determined by such extrinsic factors, the problem for natural selection is to configure an N-dimensional genome such that the resulting network interactions will map a given set of initial states to a given set of point attractor states.

An instructive example was considered by Giacomantonio and Goodhill (2010), concerning the interactions between genes Fgf8 (Assimacopoulos et al., 2012; Rakic, 2009), Emx2 (Bishop et al., 2000, 2002; Hamasaki et al., 2004), Pax6 (Hamasaki et al., 2004; Manuel et al., 2007), Coup-tf1 (Armentano et al., 2007; Borello et al., 2014), and Sp8 (Borello et al., 2014; Sahara et al., 2007), which specify position information in the embryonic neocortex and ultimately guide the growth of thalamocortical axons by chemoattraction (e.g., Sur and Rubenstein (2005); Ypsilanti and



input					output				
i	*ii*	*iii*	*iv*	*v*	*a*	*b*	*c*	*d*	*e*
0	0	0	0	0	0	1	0	1	0
0	0	0	0	1	1	0	0	1	1
0	0	0	1	0	0	0	1	1	1
0	0	0	1	1	1	0	0	1	0
0	0	1	0	0	1	0	1	1	0
0	0	1	0	1	1	1	0	1	0
0	0	1	1	0	1	1	1	1	1
0	0	1	1	1	0	1	1	0	1
0	1	0	0	0	1	1	1	0	1
0	1	0	0	1	0	1	1	0	1
0	1	0	1	0	0	1	0	1	1
0	1	0	1	1	0	0	1	0	0
0	1	1	0	0	1	0	1	0	1
0	1	1	0	1	0	0	0	0	1
0	1	1	1	0	1	0	0	1	0
0	1	1	1	1	0	1	1	0	0
1	0	0	0	0	1	0	1	0	0
1	0	0	0	1	0	1	1	0	1
1	0	0	1	0	1	1	1	1	0
1	0	0	1	1	0	1	1	1	0
1	0	1	0	0	1	1	0	1	1
1	0	1	0	1	1	0	1	1	1
1	0	1	1	0	1	0	1	0	1
1	0	1	1	1	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1
1	1	0	0	1	0	0	1	1	1
1	1	0	1	0	0	1	1	0	1
1	1	0	1	1	1	1	1	1	1
1	1	1	0	0	0	1	0	1	1
1	1	1	0	1	0	0	0	0	0
1	1	1	1	0	0	1	0	0	1
1	1	1	1	1	1	0	1	0	1

Figure 2.1: *Gene interaction network.* A network of $n = 5$ interacting genes, shown labelled a–e, each with inputs labelled i–v. The truth table determines the expression level of each gene in response to each of the $2^n = 32$ possible patterns of gene expression. The coloured elements thus constitute a 'genome' of $N = n2^n = 160$ bits, which in this case specifies a maximally fit network ($f = 1$).

Rubenstein (2016); see Greig et al. (2013); Antón-Bolaños et al. (2018) for reviews). At embryonic day 9.5 (E9.5), before the other transcription factors are known to be expressed, the telencephalic morphogen Fgf8 (Shimogori and Grove, 2005) is secreted only at the anterior neural ridge of the developing forebrain (Rakic, 2009) (see Greig et al. (2013) for a review). Together with other signalling molecules and patterning centers, the secretion of Fgf8 at E9.5 induces the graded expression of Emx2, Pax6, Coup-tf1, and Sp8 in the progenitor cells in the ventricular zone. Interactions between these genes yield posterior to anterior gradients in Emx2 and Coup-tf1 expression and anterior to posterior gradients in Pax6 and Sp8 expression. Hence, in two contexts, defined by the differential expression of Fgf8, $n = 5$ genes map initial state [00000] to the target point attractor [01010] in the posterior domain, and map initial state [10000] to the target point attractor [10101] in the anterior domain (binary expression levels ordered as the names of the genes are listed above). See Fig.2.2.

By chance mutation alone, the problem of finding a genome configuration to facilitate such developmental dynamics is very difficult. Even for $n = 3$ genes, the genome is of length $N = 3 \times 2^3 = 24$, and there are $2^N = 16777216$ possible genome configurations. An exhaustive search reveals that 11384 of these possible configurations (0.068%) map differential expression of one gene to differential expression of n genes, e.g., mapping initial states [000] and [100] to point attractors [010] and [101]. Current computing power does not allow for an equivalent figure to be determined by an exhaustive search of the genome space for more than three genes. How then has natural selection been able to discover gene-interaction networks that yield stable developmental dynamics, i.e., the emergence of point attractors?

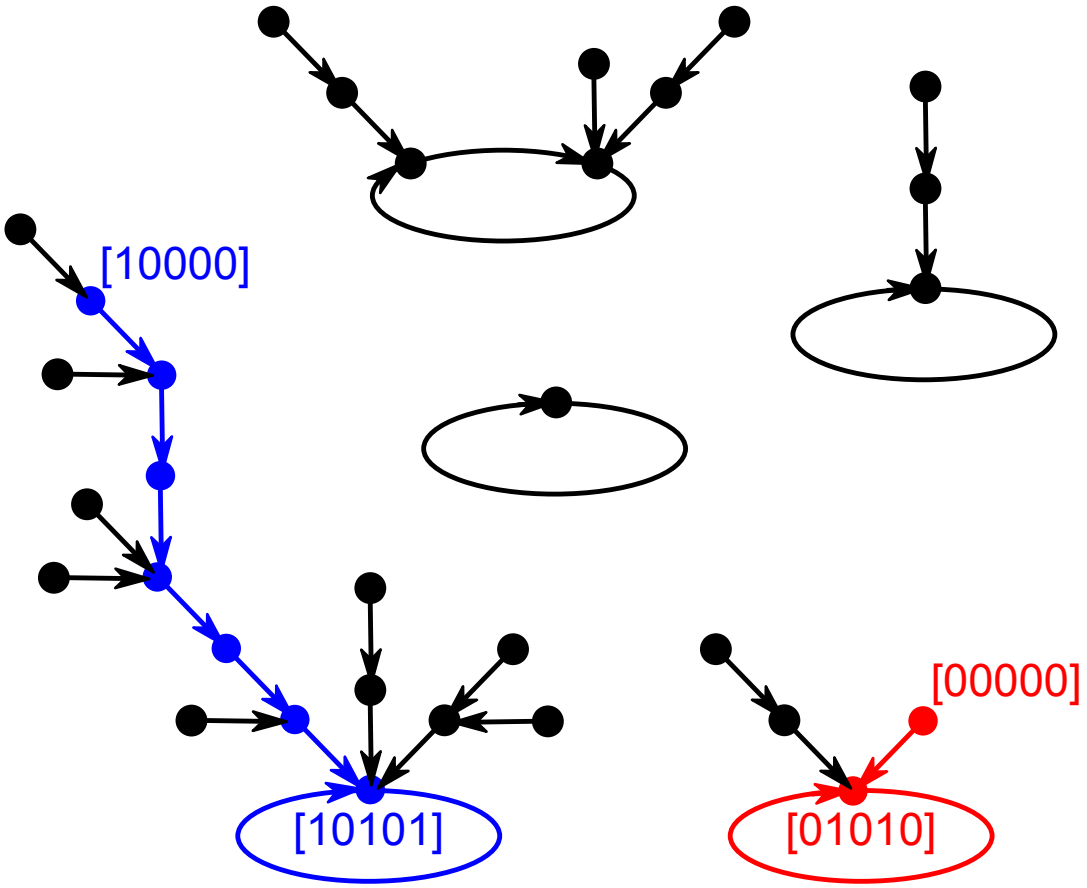


Figure 2.2: *Attractor landscape.* The developmental dynamics of the network that is specified by the genome in Fig. 1 reveals five attractors (four point attractors and one with a limit cycle of length two). Every possible gene expression pattern is represented by one dot, and the transitions between states are represented by arrows. Initial states [10000] and [00000] map to target states [10101] and [01010] as point attractors. The blue path corresponds to the development of the network in the anterior context and the red path corresponds to the development of the network in the posterior context.

2.2 Results

2.2.1 Modelling the interaction between self-organization and selection

Discovering the genetic conditions from which specific point attractors can emerge becomes tractable, even for larger networks, if we assume that self-organization, i.e., the emergence of attractor dynamics during development, is able to interact with natural selection in the following way. Starting with a random genome, a network of $n = 5$ genes ($N = 160$) might from some initial state, e.g., [00000], settle into a particular limit cycle, e.g., [11000] then [00011] then [01011], before repeating [11000] and continuing indefinitely. As the network continues to cycle through these three states, the five genes will be expressed for the following proportions of time: 1/3, 2/3, 0/3, 2/3, 2/3. These values correspond to the relative production rates of five proteins. If the target point attractor state is e.g., [01010], then the protein production levels are 'correct' in the following proportions: 2/3, 2/3, 3/3, 2/3, 1/3. The mathematical product of these values thus represents the extent to which downstream processes will be orchestrated by the correct distribution of proteins, and as such it can be used as a measure of the fitness of the genome,

$$f = \prod_{x}^{X} \prod_{g}^{n} \left(1 - \frac{1}{T} \sum_{t}^{T} |s'_{g,x} - s_{g,x,t}| \right), \tag{2.1}$$

where x is an index over X contexts (each defined by an initial network state), t is an index over T states comprising the attractor for context x, and s' is the target level of gene expression.

In simulation, states s can be identified simply by iterating the network dynamics 2^n times (to guarantee that an attractor is reached) and then iterating a further T times until a repetition is detected. Natural selection can then be represented in its simplest terms by flipping each of the N genome bits with probability $p \in [0, 0.5)$, accepting the modified genome if $\Delta f \geq 0$, and repeating the process for each simulated generation.

It is important to emphasise that according to equation (1), the fitness is derived from the developmental dynamics of a network by comparing the target state with the *time average* expression level of each gene, given that the network will cycle through the states in its attractor indefinitely. Hence the model represents an assumption that dynamics in gene networks can propagate quickly with respect to the timescale over which the corresponding protein levels accumulate and interact with downstream processes (see Shimojo et al. (2008); Imayoshi et al. (2013)). Note that according to equation (1), $f = 1$ only if all initial states map to the target states as point attractors. Note also that $f = 0$ if any gene (g) is expressed incorrectly in any attractor state (s) that is visited in any context (x). Furthermore, note that none of the states in the limit cycle are required to correspond exactly to the target state for the network to be considered to have positive fitness, as is the case for the example considered above. Finally, note that setting $p = 0.5$ defines a control condition in which the evolutionary process becomes equivalent to a random sampling of points in the genome space.

2.2.2 Limit cycle dynamics can guide evolution

We consider first the example from neocortical development ($n = 5$, $X = 2$), where initial state [00000] should map to state [01010] as a point attractor, and initial state [10000] should map to state [10101] as a point attractor. Evolving for a total of 10^8 simulated generations at each mutation rate, from $p = 0.05$ to $p = 0.45$ at increments of 0.05, revealed dynamics similar to 'punctuated equilibria' (Gould and Eldredge, 1977; Bak, 1996), whereby long periods of stasis ($\Delta f = 0$) were punctuated by increments in fitness (see Fig. 2.3). The distribution of the number of generations in each period of stasis is shown in Fig. 2.9.

At each mutation rate, the distribution of the number of generations required to discover a maximally fit genome ($f = 1$) was long-tailed, conforming increasingly to a log-normal distribution for smaller values of p, i.e., for an increasingly local search of the genome space (see Fig. 4). At $p = 0.05$, the number of $f = 1$ genomes discovered was 70 times greater than by random sampling ($p = 0.5$), with discovery taking 853 generations on average. Reducing the mutation rate further (i.e., flipping an average of 6 or less bits per generation) reduced the evolutionary speed-up, confirming that the fitness landscape is not smooth near the fitness peaks. Overall, the average number of generations required to discover $f = 1$ networks, μ, was well approximated by $\mu(p) = e^{8.90p+6.22}$ for $p > 0.05$ (for values of p less than 0.05, $\mu \to \infty$ as $p \to 1/N$ and $\mu(0) = \infty$).

These results show that self-organization can accelerate selection under the assumption that dynamically stable protein production levels that are more similar to ideal levels yield better fitness. Under this assumption, approximate network solutions that emerge within a lifetime as limit cycle attractors can provide a scaffold of

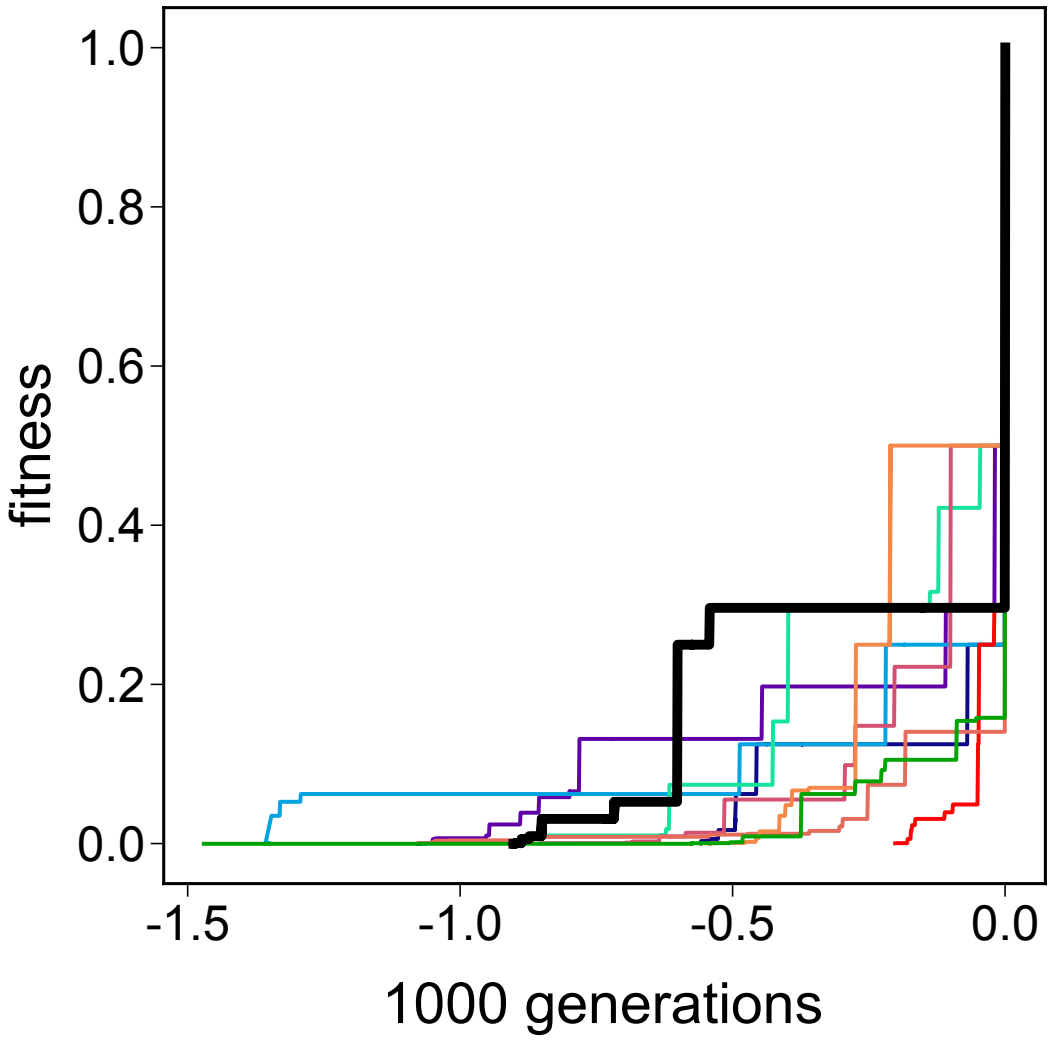


Figure 2.3: *Punctuated equilibria.* Evolution of ten genomes by attractor scaffolding (mutation rate $p = 0.1$), with the generations at which $f = 1$ aligned to zero. Evolution yields long periods of stasis punctuated by sharp fitness increments. The bold trace shows the evolution that gave rise to the network detailed in Figs. 2.1 and 2.2.

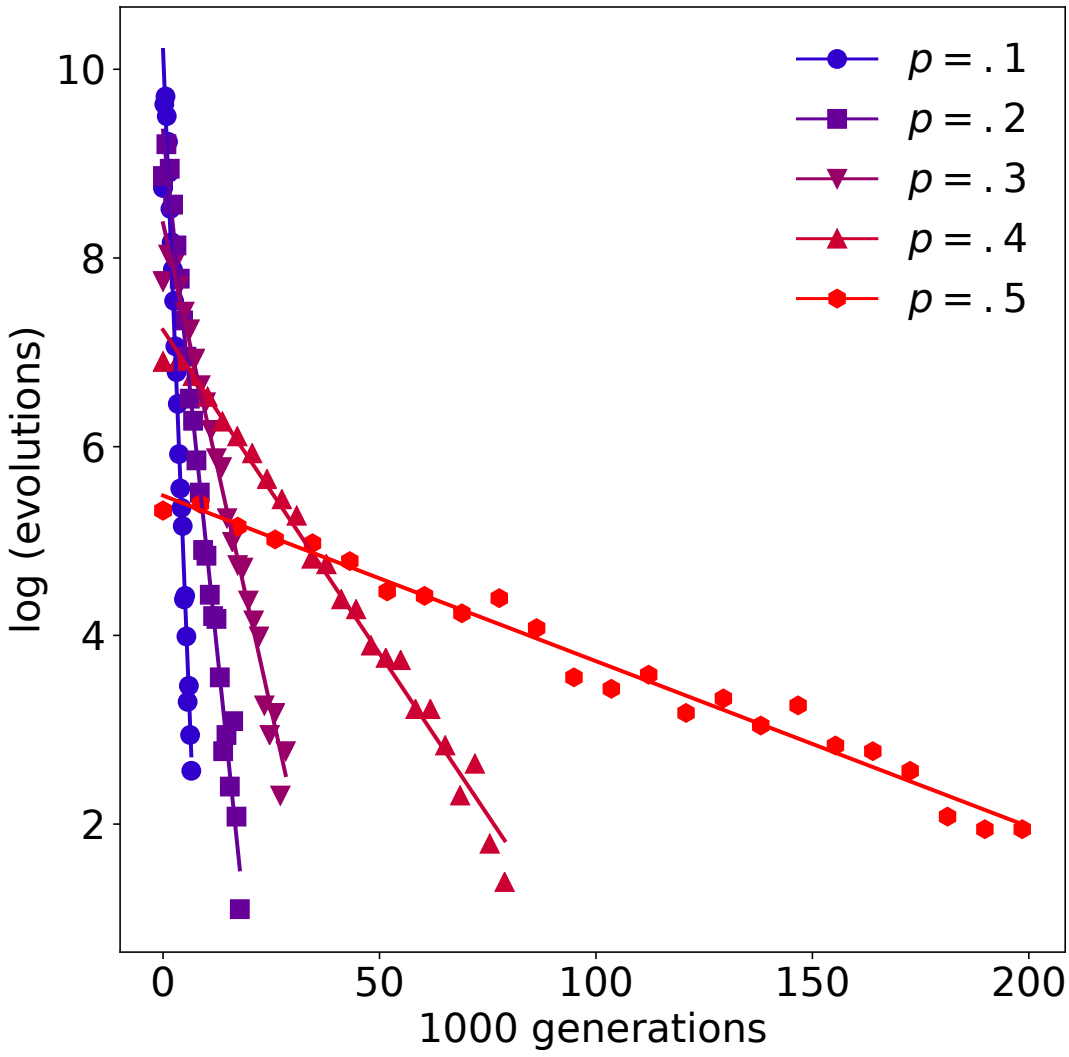


Figure 2.4: *Limit cycle dynamics guide evolution.* Distribution of generations required to discover $f = 1$ networks at a range of mutation rates p. Evolution at lower mutation rates corresponds to searching the fitness landscape more locally, and is shown here to increasingly accelerate the discovery of maximally fit $f = 1$ networks. Note that $p = 0.5$ corresponds to a random search.

graded fitness around otherwise isolated peaks in the fitness landscape, for natural selection to climb. We thus refer to this mechanism as *attractor scaffolding*. Self-organization can only assist selection via attractor scaffolding if the embedding of attractor landscapes in the N-dimensional genome space is locally structured, as is evidenced here by further accelerations in the discovery of $f = 1$ genomes at lower mutation rates, i.e., as the search through genome space is more local.

2.2.3 Fit networks and random networks have equal complexity

Many known biological networks comprise Boolean functions that belong to particular classes of low complexity, such as threshold functions and canalizing functions, i.e., genes for which either expression level is guaranteed by a specific expression level (1 or 0) in at least one other gene (Kauffman, 1969, 1993; Szejka and Drossel, 2007; Paul et al., 2006). To investigate whether networks generated by attractor scaffolding belong to such classes, we compared the complexity of functions generated in ten thousand $f = 1$ networks to the complexity of functions generated in ten thousand randomly configured networks. Following Gherardi and Rotondo (2016), we used the Quine-McCluskey algorithm to derive, for each gene, an equivalent Boolean logic expression (in disjunctive normal form) with the fewest terms, and measured the complexity of each as the number of terms normalised by 2^n. A t-test revealed no significant difference between the mean complexity of Boolean functions in $f = 1$ networks (mean=0.23$\pm$0.017) and in randomly generated networks (mean=0.23$\pm$0.017). As we might therefore expect, very few fit networks (0.11%), like random networks (0.11%), contained canalizing functions, which are typically observed in $n = 5$ net-

works for complexities of around 0.1 (Gherardi and Rotondo, 2016). There was also no difference between fit and random networks in terms of the bias, i.e., the proportion of truth table values that are 1 (mean=0.50 ± 0.04 versus 0.50 ± 0.04), or the number of attractors (mean=2.46 ± 1.12 versus mean=2.43 ± 1.12). Hence we conclude that networks generated by attractor scaffolding are equal in complexity to randomly configured networks.

2.2.4 Attractor scaffolding is robust to the choice of contexts and integration method

We next investigated whether attractor scaffolding is sensitive to the choice of the initial and target states that define each context. In the example from neocortical development, the initial states differed by one bit (the expression level of the first gene, representing Fgf8) and the target states differed by n bits (i.e., defining n binary gradients). So we first repeated the original simulations using an anterior initial state that differed from the posterior initial state by a Hamming distance that ranged from 1 (e.g., state [10000] as used originally) to n (e.g., [11111]). The choice of the initial state had no effect on the evolutionary dynamics. Next we repeated the original simulations using an anterior target state that differed from the posterior target state by a Hamming distance that ranged from 1 (e.g., state [11010]) to n (e.g., [10101] as used originally). Fig.2.5 shows how attractor scaffolding depends on the distance between the two target states. The number of generations required to discover $f = 1$ networks was found to decrease as the Hamming distance between the two target states was increased, with networks mapping from initial states that differed by one bit to target states that differed by one bit discovered in 3574 generations on average,

corresponding to an evolutionary speed-up of 16. Note that the choice of target state had no effect on the distribution of discovery times in the $p = 0.5$ control condition.

The evolutionary speed-up was also affected by the number of contexts, X. For an arbitrary choice of initial and target states (all unique), we ran simulations for $X = 2$, $X = 3$, and $X = 4$ contexts. The simulation was evolved at a range of mutation rates ($p < 0.5$) until 1000 networks were discovered for each combination of X and p. As the number of contexts was increased, more generations were required to discover fit genomes (see Fig.2.6). To obtain sufficient data for analysis, the model was run for the $p = 0.5$ control condition through 10^8 ($X = 2$), 5×10^9 ($X = 3$), and 2×10^{11} ($X = 4$) generations. Although the shortest $f = 1$ discovery period increased with X, the discovery period in the $p = 0.5$ control case increased at a far greater rate. Defined as the mean discovery period at $p = 0.5$ (where periods were always maximal) divided by the mean discovery period at $p = 0.05$ (where periods were always minimal), the evolutionary speed-up by attractor scaffolding *increased* with the number of contexts. Remarkably, the speed-up increased at an approximately exponential rate with the number of contexts (Fig.2.7). Thus, as the set of target attractors increased in size, the effectiveness of attractor scaffolding was maintained, despite an exponential increase in the difficulty of the search.

As a final test of robustness, we adapted the original simulation (neocortical example with $X = 2$) to instead update the state of the network asynchronously. Instead of changing the states of all n genes at the current timestep based on the states of all n genes at the previous timestep, asynchronous updating involves iteratively changing the state of one randomly selected gene at a time. To compute the fitness, these dynamics were iterated $T = 2N$ times from the initial state in each context and equation (1) was calculated with respect to all XT visited states; an evolutionary

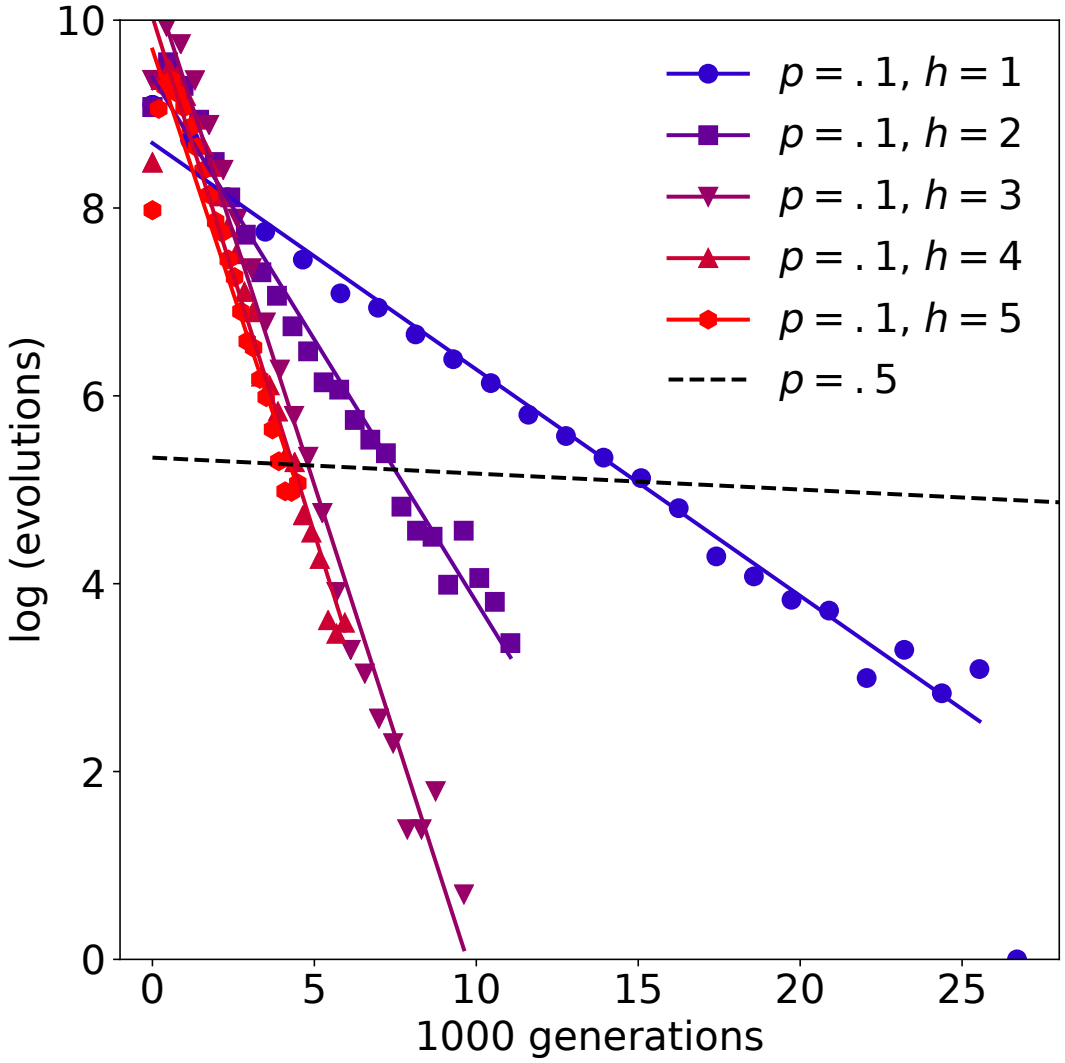


Figure 2.5: *Attractor scaffolding varies with distance between targets.* The number of generations required to discover maximally fit networks decreases as the Hamming distance (h) between $X = 2$ target states increases. Solid lines show fits for distributions obtained using a mutation rate of $p = 0.1$. Distributions for each h were identical for the $p = 0.5$ control (dashed line). The minimum evolutionary speed-up was a factor of 16 (comparing the mean evolutions at $p = 0.1$ and $p = 0.5$ for $h = 1$).

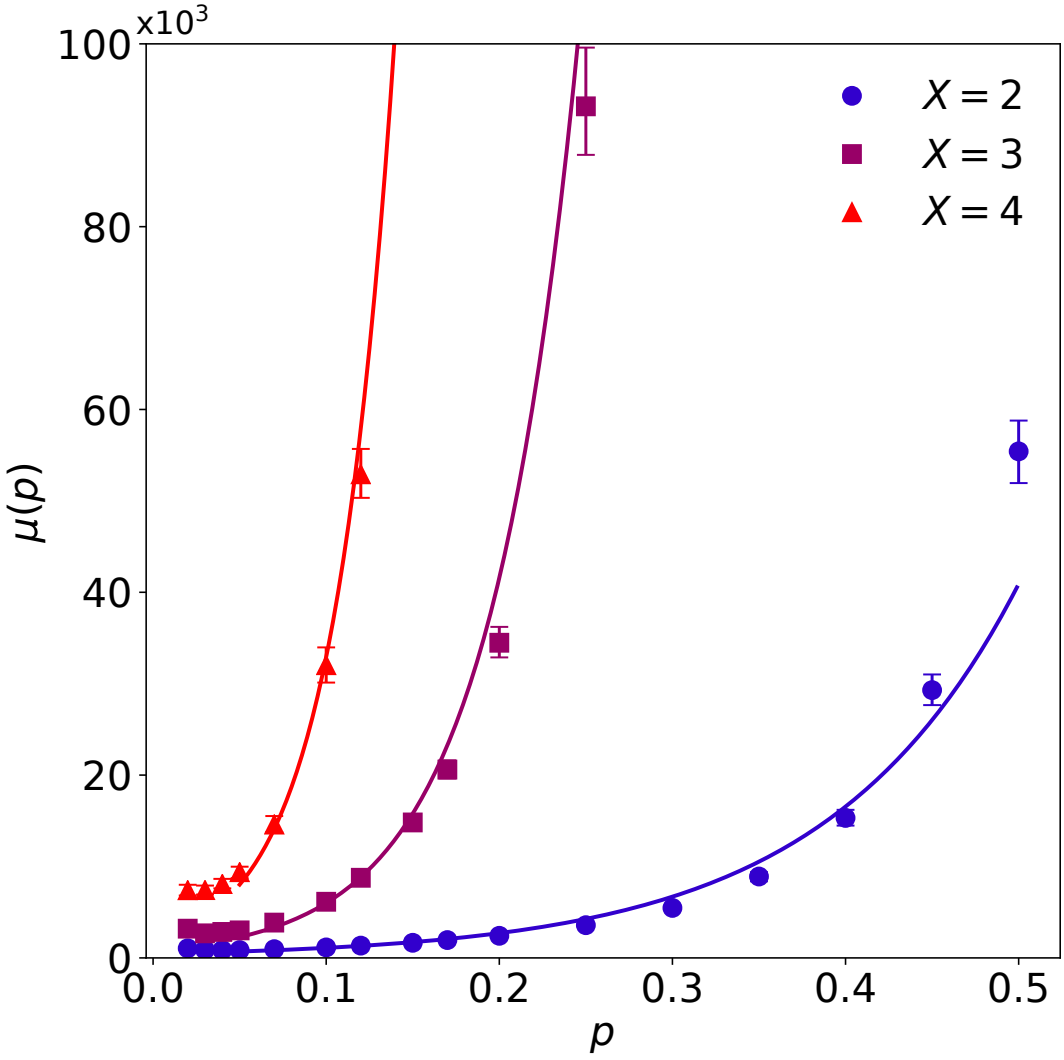


Figure 2.6: *Attractor scaffolding varies with the number of contexts.* As the number of contexts X increases, more generations are required to discover fit genomes. For each value of X, the mean discovery period μ increased approximately exponentially with the mutation rate p (fits shown by solid lines). Error bars show 95% confidence intervals, determined by a bootstrap analysis of the mean, μ, with 1024 resamples.

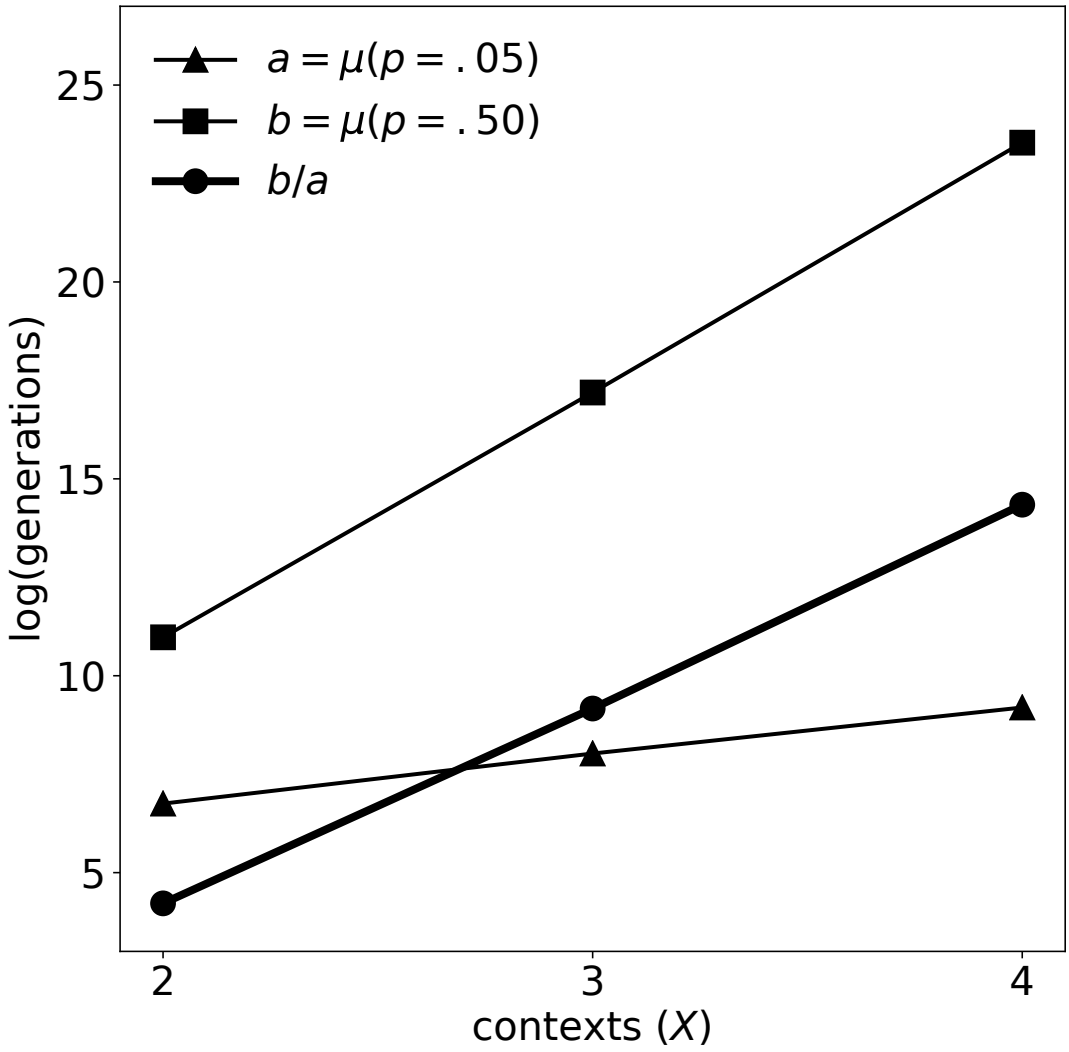


Figure 2.7: *Evolutionary speed-up increases with the number of targets.* The maximum average discovery period b (for the $p = 0.5$ control condition), divided by the minimum average discovery period a (for $p = 0.05$) yields an estimate of the evolutionary speed-up by attractor scaffolding (circles connected by a thick line), and is here shown to increase exponentially with the number of contexts X.

run was terminated when $f > 0.95$. Overall, the average number of generations, μ, required to discover fit networks was well approximated by $\mu(p) = e^{13.60p+5.33}$ for $p > 0.15$ (for values of p less than 0.15, $\mu \to \infty$ as $p \to 1/N$ and $\mu(0) = \infty$) (Fig.2.8). The effect of attractor scaffolding is therefore robust when the implicit assumptions of a synchronous clock and deterministic network interactions are relaxed.

2.3 Discussion

The effect of self-organization by attractor scaffolding resembles that presented by Hinton and Nolan (1987). In their seminal model, some genome bits are adaptable within the lifetime of each member of a population, and their genomes are recombined with a probability that decreases with the number of flips of these adaptable bits before a target genome is discovered. The state of adaptable bits is not inherited, but inheritance of the ability to flip state nevertheless increases the discovery rate. Faster discovery of target states within lifetimes therefore directs selection pressure in favour of genetic conditions from which targets can be more rapidly acquired. Attractor scaffolding confers a similar advantage; in both cases an approximation to the target is maintained within the lifetime and communicated only indirectly between lifetimes. An important distinction is that by attractor scaffolding, the benefit of distributing approximate solutions across limit cycle states, rather than across members of a population, is conferred by developmental dynamics intrinsic to individual organisms.

It is interesting that attractor scaffolding was affected by the similarity between the target states. One implication of this result is that networks like that described

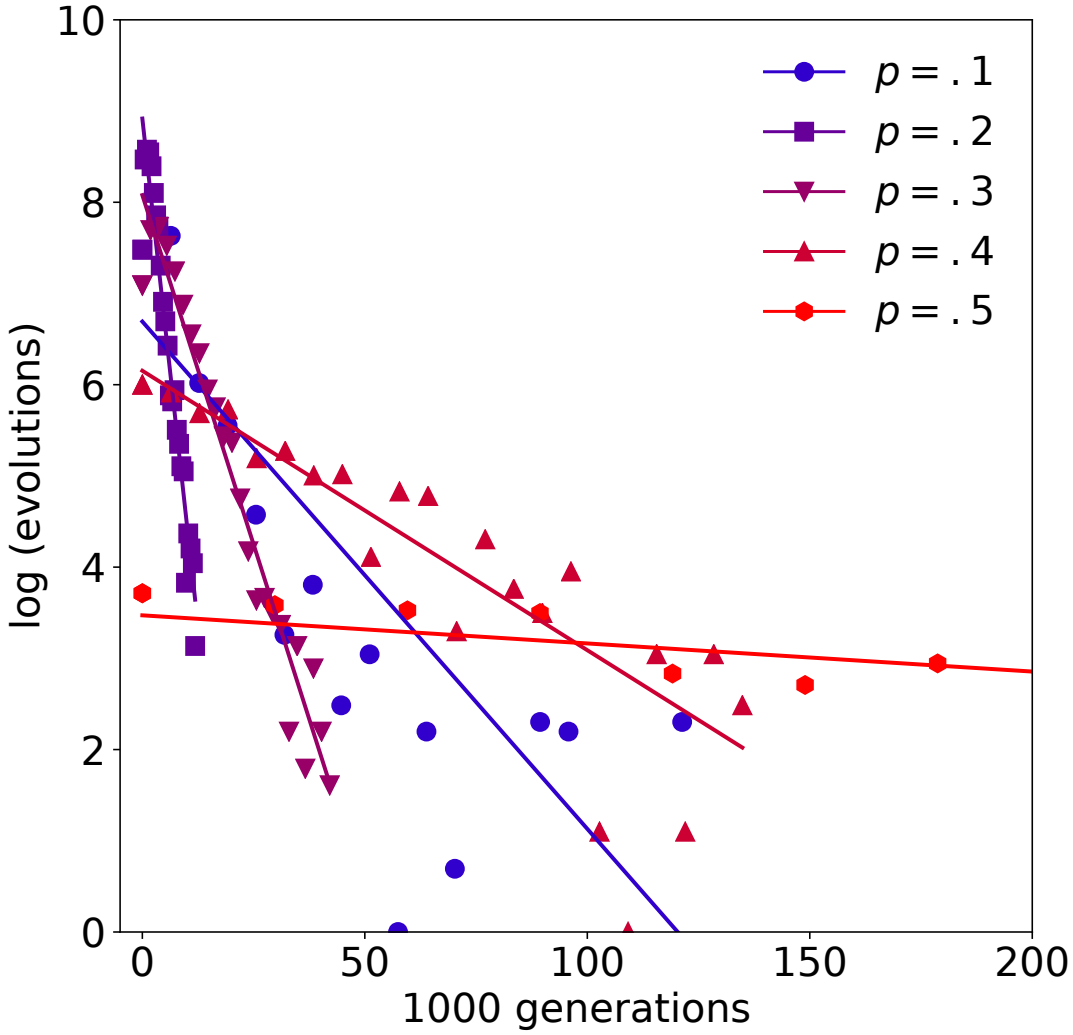


Figure 2.8: *Asynchronous updating.* The simulation used to create Fig.2.4 was modified to update the state of the network asynchronously. The trend for faster discovery of fit networks at lower mutation rates was comparable, though note that the maximum evolutionary speed-up was obtained at $p = 0.15$. Attractor scaffolding is therefore robust to the choice of integration method.

for cortical arealization, which can be perturbed by the expression of a morphogen into generating orthogonal patterns of stable gene expression, may be more likely to be discovered by natural selection. It is also interesting that the evolutionary speed-up via attractor scaffolding increases exponentially with the number of target attractors. This result suggests that the challenge for an evolutionary search based on iterative evaluation of chance mutations may scale with the complexity of the phenotype (i.e., with properties of the emergent attractor landscape) rather than with the complexity of the genotype (i.e., with the naturally occurring frequency of fit network specifications).

A criticism of this model that could be raised is that the model genome and its mutations do not correspond closely enough with those of real organisms. Our model genome determines the interaction of 5 genes, described by $n2^n = 160$ 'bases' which are 0 or 1. This creates a genotype space of 2^{160} genomes which are the vertices of a 160 dimensional hypercube. So there are an enormous number of possible genomes, but they are all very close to each other mutationally. Combine this with the fact that the number of distinct phenotypes is actually quite small, and we have an explanation for how the evolutionary algorithm can be so successful. Is it fair to extend this argument to real genomes? Are they also highly dimensional and compact, leading to a small number of attractor basins? The other chapters in this thesis address this issue by adapting the model to incorporate more qualities of real genomes, but let us now summarise what those are. A similar 5 gene network in biology would have tens of thousands of bases, not 160. Mutations would be more likely to cause small continuous shifts in gene expression, not discrete jumps. The number of attractor basins is also relatively small in real genomes, this can be seen by the many different genomes that lead to qualitatively similar phenotypes. So I would argue that the

key characteristics of these systems are comparable. Essentially, a small number of mutations can widely explore the genotype space, and yet lead to a small set of phenotypes.

Attractor scaffolding offers a potential mechanism for genetic assimilation; by the gradual evolution of limit cycle dynamics towards point attractor dynamics. Thus, it might support a range of epigenetic phenomena, such as the Baldwin effect(s) (Baldwin, 1896; Weber and Depew, 2003; Dennett, 2003; Deacon, 2003). Similar principles may help to explain the 'molecular logic' of specific gene networks, e.g., networks responsible for the embryonic development of neocortical circuits (Giacomantonio and Goodhill, 2010; Greig et al., 2013; Wolf, 2005; Balaskas et al., 2012), and how intrinsic properties of network self-organisation may similarly constrain the evolution and development of functional neuronal networks (Harris et al., 2002; Kaschube et al., 2010; Wilson and Bednar, 2015; Krubitzer and Prescott, 2018). Practical applications may involve new methods for programming large circuits of logic operations. For example, we found that an $n = 7$ circuit, for which the space of configurations comprises $N = 2^{896}$ possibilities, can be configured to robustly map three initial states to three distinct target states in a few million computationally inexpensive steps.

2.4 Methods

A standalone implementation of the model is provided as Supplementary Material S2 and a script for recreating Fig.2.4 from the main text is provided as Supplementary Material S3. Copy the text from S2 into a file with a .cpp extension, e.g., evolve.cpp,

and copy the text from S3 into a file with a .py extension, e.g., plot.py. From the command line compile using e.g., 'g++ -O3 evolve.cpp -o evolve', run the model using './evolve', then plot using 'python plot.py'. These programs are part of a full repository of code and additional analysis and visualization tools maintained at https://github.com/ABRG-Models/AttractorScaffolding.

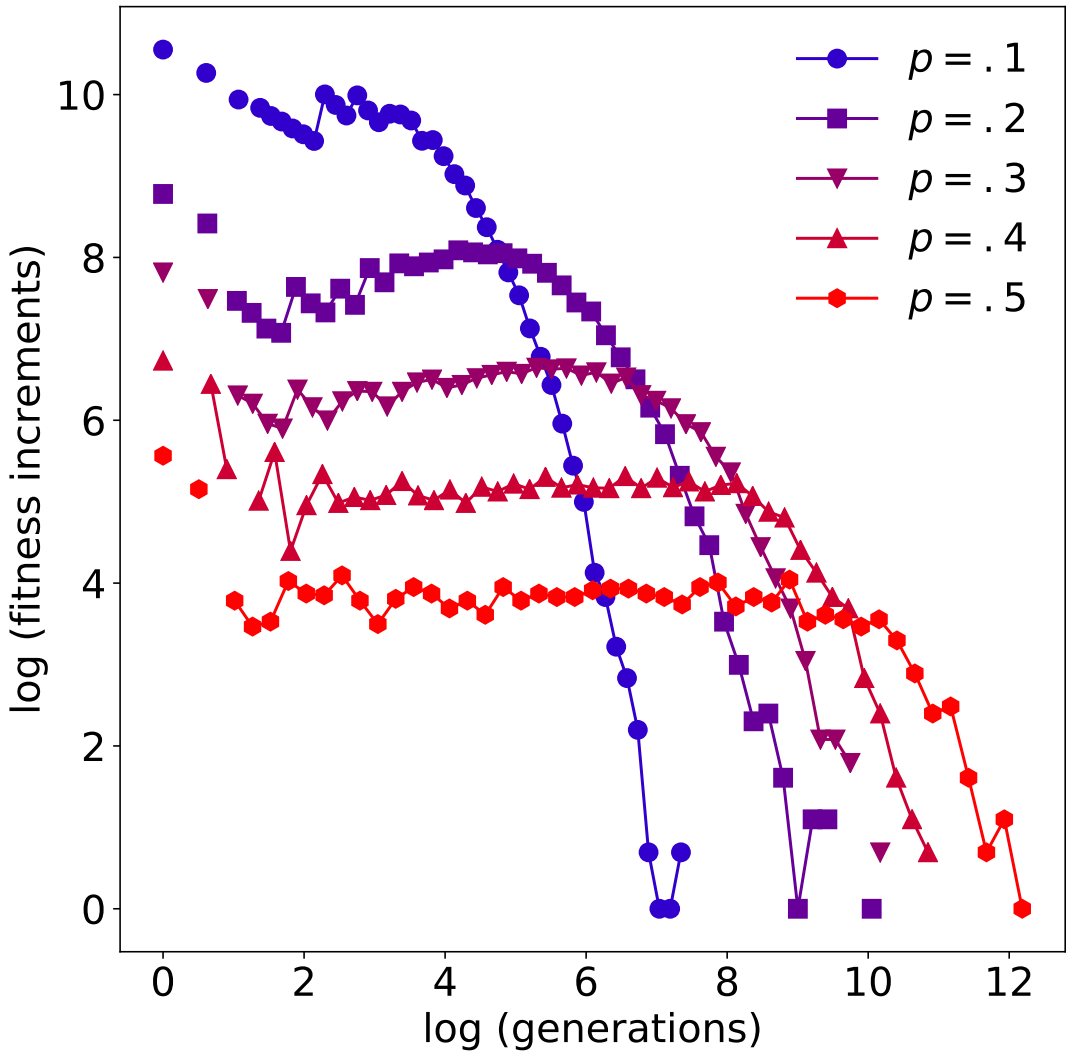


Figure 2.9: *Punctuated equilibria.* The distribution of the number of generations in each period of stasis. For lower values of p, there were fewer generations of stasis between each fitness increment.

Chapter 3

Reverse-engineering gene-interaction networks from spatial patterns of expression

Boolean Networks have given a useful framework for investigating evolvability, but the complexity of spatial patterns is extremely limited compared to real cortical phenotypes. Here we develop a new model of gene regulation in order to overcome this limitation.

Genes interact in complex networks that in many important ways resemble the interactions between neurons in brain networks, and so the techniques from neural network theory should also be applicable to gene networks. While the mathematical relationships between genes and neurons have long been understood, surprisingly few have used the most fundamental tools of neural network theory, learning algorithms,

to understand gene networks. Here we show how such tools can be used to reverse-engineer gene networks, through the specific example of the gene network responsible for dividing the cortex into different information-processing areas.

3.1 Introduction

Changes in gene expression patterns observed in knockout experiments provide important clues about the structure of the gene networks responsible for dividing tissues into distinct functional areas. Insights gained from such experiments are often formulated in qualitative terms — regions may 'shrink' or 'grow', or be 'displaced' or 'distorted'. While these effects may be individually quantifiable, a more principled approach may allow insights from multiple experiments to be combined. Here we develop such an approach and show how changes in expression patterns caused by manipulations of individual genes can be analysed in combination to reverse-engineer an underlying gene network.

The positional information model of Wolpert (1969, 1996) has been influential as a description of the mechanisms by which spatial patterns of gene expression are determined by interactions between genes. Accordingly, a specific area of a tissue can be identified when the expression of a given gene is determined by the application of a threshold to the graded expression of one or more others (Ashe, 2006; Rogers and Schier, 2011). The textbook example is the patterning of the drosophila embryo into striped segments, by a combination of thresholding operations applied to the levels of expression of genes such as Hunchback, Giant, and Kruppel, and with respect to an overall anterior-posterior gradient in the expression of morphogen Bicoid (Driever

and Nüsslein-Volhard, 1988; Frigerio et al., 1986). Definite regions are then identified by markers such as Eve, which is expressed in alternating stripes, each defined by a particular combination of thresholds of the enhancers upstream, that are determined by the configuration of cis-regulatory module binding sites (Jaeger and Martinez-Arias, 2009).

Arbitrarily complex two-dimensional patterns can also be established in this way, by layering expression gradients and thresholding operations. For example, the emergence of distinct fields for processing visual, somatosensory, auditory, and motor information in the developing mammalian neocortex has been described in terms of a sequence of Boolean logic operations within a network of approximately 20-40 genes (Greig et al., 2013; Rakic et al., 2009). The sequence is triggered by the graded expression of morphogens (Wnts, BMPs, FGFs) from specific 'patterning center' locations on the perimeter of the subventricular zone (Grove and Fukuchi-Shimogori, 2003), and the subsequent expression of transcription factors Emx2 (Bishop et al., 2000, 2002; Hamasaki et al., 2004), Pax6 (Hamasaki et al., 2004; Manuel et al., 2007; Ypsilanti and Rubenstein, 2016), Sp8 (Borello et al., 2014; Sahara et al., 2007), and Coup-tf1 (Armentano et al., 2007; Borello et al., 2014) in pairs of orthogonal gradients (Walther and Gruss, 1991; Gulisano et al., 1996; Waclaw, 2006; Zhou et al., 2001; Liu et al., 2000). These patterns subsequently give rise to species-specific shapes in the expression of 'area marker' genes such as Id2 and RZRβ (Dye et al., 2010a,b), which guide thalamocortical growth and innervation, and ultimately specify cortical fields with similar shapes (Rakic, 2009; Rakic et al., 2009; Greig et al., 2013; Kast and Levitt, 2019). Specific changes in the shapes and positions of the cortical fields can be caused by knocking out any of these genes (see Fig. 3.2, top).

As the number of genes in a network grows, and the complexity of the spatial

patterns that it can produce increases, the possible ways in which the Boolean logic operations can be configured grows exponentially. For example, making the simplifying assumption that expression levels are binary, considering expression levels in only two locations (the anterior or posterior half of the tissue), and describing only the interactions amongst five genes (morphogen Fgf8; (Shimogori and Grove, 2005; Assimacopoulos et al., 2012), and the transcription factors listed above), there are a total 1.68×10^7 possible Boolean networks, of which approximately one in a thousand will create the correct patterns (Giacomantonio and Goodhill, 2010; Wilson et al., 2019). For a network like that for cortical patterning, which consists of dozens of genes, the number of combinations of Boolean operations that is compatible with the available data quickly becomes astronomical. How then can we use the positional information model to derive from observed spatial patterns of gene expression, new knowledge about the structure of the gene networks that gave rise to those patterns?

To address this question, consider that threshold units have been the basic building blocks of artificial neuron models since McCulloch & Pitts first described their properties around eighty years ago (McCulloch and Pitts, 1943). The crucial idea was that a threshold for activation will be exceeded when the pattern of activation levels presented as input to the unit is aligned to the pattern of corresponding connection weights, such that the output of the unit serves to detect a specific input pattern (Rosenblatt, 1958). Around thirty years ago Rumelhart and colleagues derived the backpropagation algorithm – a method for finding a set of connection strengths in a feedforward network that produces a given input-output mapping (Rumelhart et al., 1986). This method is based on gradient descent, i.e., making incremental adjustments to an initially random choice of connection strengths that are directed by the changing gradient of the discrepancy between current and target responses. Thus

it requires the threshold function of each unit to be differentiable, with the sigmoid function, $y(x) = (1 + e^{-x})^{-1}$, being a popular choice. A decade later, an important paper by Bray made a compelling case for modelling gene interactions using threshold units, with ligand binding affinities substituting for synaptic weights, and sigmoidal ligand-binding probability functions substituting for the neuronal transfer function (Bray, 1995). But for the specific historical contexts in which the particular analogies were invented, the scientific exploration of networks of threshold units might just have easily become known as 'artificial gene networks' instead of 'artificial neural networks'. In the words of Bray "As with neurons, so with proteins." ((Bray, 1995); pp. 308).

While the analogy between neural networks and gene networks has been fruitful in terms of simulating and analysing gene network dynamics and revealing common motifs in gene network structure (e.g., Vohradsky (2001); Buchler et al. (2003)), surprisingly little research has been directed at using the most important set of techniques developed through neural networks research; learning. Supervised learning algorithms like backpropagation consolidate the structure of a given task in the pattern of connection weights amongst network nodes. As such, each pattern of learnt connection weights constitutes an internally consistent hypothesis about the configuration of the biological network that carries out that mapping, whose validity can be measured in terms of the reconstruction error. The key idea that we explore in this paper is that such hypotheses can be formulated in increasingly specific detail by incorporating additional constraints on the learning process, for example by training artificial gene networks to produce both wildtype and pathological patterns observed in gene knock-in or knock-out experiments.

A variety of experimental manipulations to the expression patterns of the genes

involved in cortical arealization have been shown to shift, shrink, stretch, skew, ablate, combine, or even duplicate cortical fields (see Greig et al. (2013); Assimacopoulos et al. (2012)). Using this approach, we therefore set out to address some basic questions about the structure of the cortical arealization network as a worked example: How complex is the network, i.e., how many threshold operations are required for successful pattern reconstruction? What kind of network topology is most compatible with the available data on spatial patterning? What might the structure of reverse-engineered networks reveal about the development of tissue patterning?

3.2 Results

Reverse engineering gene networks for neocortical arealization

Artificial gene networks were trained using the recurrent backpropagation algorithm introduced by Pineda (1987, 1989), which is conceptually similar to the backpropagation algorithm (Rumelhart et al., 1986), but has the advantage that it may be applied to train networks of arbitrary topology. We consider two general types of network architecture; feedforward networks with all-to-all connectivity between successive layers, and fully recurrent networks (see Fig.3.1). Both capture relevant aspects of biological gene networks – feedforward nets with sufficient numbers of hidden nodes can approximate any smooth function with relatively few connections (Hornik et al., 1989), whereas all possible network topologies may be considered special cases of fully recurrent networks when connection strengths are chosen appropriately.

Training data were extracted from images of cortical arealization patterns in the wildtype mouse, and in Emx2, Coup-tf1, and Pax6 knockouts, by Greig and colleagues (Greig et al. (2013), their Figure 3). These images depict how the shapes and positions of the primary visual (V1), somatosensory (S1), auditory (A1), and motor (M1) cortices, vary between conditions. They are not primary data, however as expert renderings in a common reference frame they nicely capture the known variations in arealization pattern and the essential spatial logic of the available data, at the resolution of statements such as "in Coup-tf1 knockout mice, M1 engulfs the region otherwise occupied by A1, S1, and V1, causing an anterior shift and a reduction in the size of the primary sensory regions" (see Fig. 3.2). The challenge for the learning algorithm is effectively to extract and combine such logical statements across wildtype and knock-out conditions, and to re-represent this information in the learnt structure of the network connections. It should be noted that we are in no way modelling the biological process of how a successful network comes to be. The algorithm is used to find successful networks so that their characteristics can be statistically analysed, in a completely different way to how biology finds these successful networks (i.e evolution). So it is no problem that we are using supervised learning based on multiple knockout training sets at the same time, using information that isn't available to the organism.

All networks consisted of at least nine nodes; two input nodes whose input values represent coordinates on the x-y–plane of the tissue, four output nodes corresponding to markers for V1, S1, A1, and M1, and three hidden nodes that we identify with Coup-tf1, Emx2, and Pax6, and refer to as the 'knockout' nodes. We treat the number of additional hidden nodes as an independent variable, against which to evaluate the ability of various network topologies to recreate the wildtype and

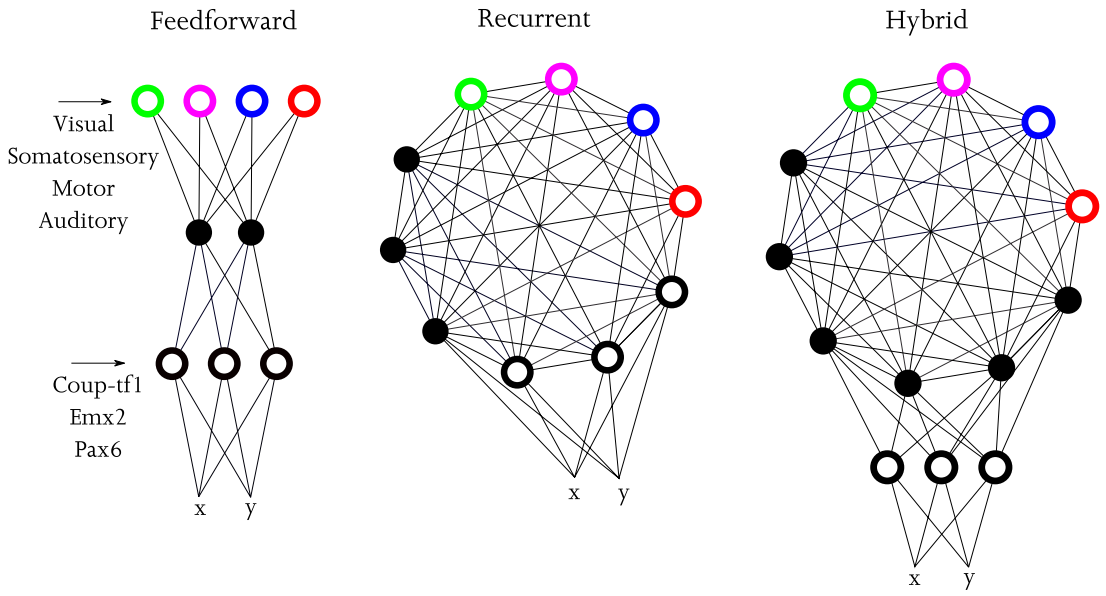


Figure 3.1: *Structures of model networks.* Each network takes x and y coordinates of the model cortex, weighted and combined through various nodes to give values at the output nodes (coloured). The Feedforward Network is the common perceptron, the recurrent and hybrid networks include cyclic connections which requires the node values are settled over several iterations until convergence. The 'knockout' nodes (Coup-tf1 etc) can be set to zero, simultaneously the target outputs are changed to match the training set for that knockout.

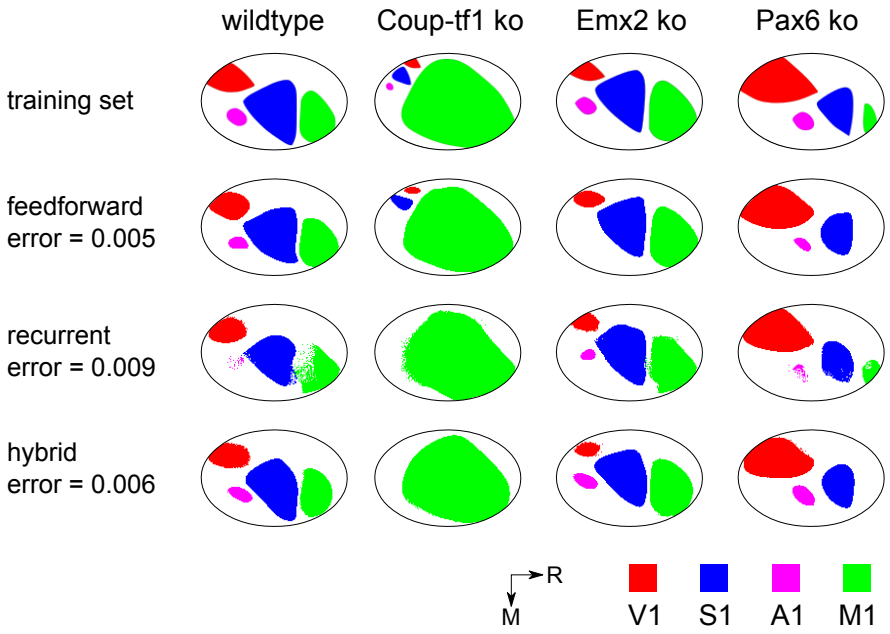


Figure 3.2: *Reverse engineering gene interaction networks for neocortical arealization.* Artificial gene networks were trained to reproduce the four patterns of neocortical arealization shown in the top row, which show the shapes and arrangement of the primary sensory and motor fields of wildtype mice, and how these patterns differ in Coup-tf1, Emx2, and Pax6 knockouts. Arealization patterns were reconstructed using a feedforward network (second row), a fully recurrent network (third row), and a hybrid network (fourth row; see text for details), based on the settled responses of output nodes representing markers for the primary visual (red), somatosensory (blue), auditory (pink), and motor (green) areas. Knockout patterns were generated from these three networks by fixing the expression of corresponding nodes to zero. Excellent reconstructions could be obtained using each architecture. These three example networks each consisted of 17 nodes, and produced reconstruction errors of approximately the level that we used to characterize reconstructions as 'good'. Top row adapted from Greig et al. (2013), their Figure 3.

knockout images. All networks were trained on five million patterns, each pattern specifying the target cortical field identity at a randomly chosen coordinate on the cortex, with identities sampled from the wildtype and three knockout images in equal proportions. Crucially, the activation of a knockout node was set to zero when training data were selected from images of the corresponding knockout pattern. After training, networks were evaluated by presenting 1000 pairs of input values (i.e., 1000 coordinates on the cortical sheet), disabling none of the nodes for a quarter of those coordinates (simulating the wildtype condition) or disabling one of the three knockout nodes for the remaining three quarters. At each coordinate the network dynamics were allowed to settle, and the discrepancy was measured between the responses of the four output nodes and their target responses, as determined by the cortical field identity at that location in the corresponding wildtype/knockout image. Finally, the reconstruction error was calculated by squaring the discrepancy and averaging it over the four output nodes and the 1000 coordinates tested (see *Methods*).

Visual inspection of the recreated arealization patterns revealed that even small networks of either topology were able to simultaneously represent the essential features of the wildtype and knockout patterns (see Fig. 3.2). While the approach cannot definitively exclude alternative possibilities, this basic result confirms that the genes that interact to specify neocortical field shapes and locations may do so by weighting and thresholding the expression levels of certain combinations of other genes.

3.2.1 Estimating the complexity of gene networks

To estimate the complexity of the network for cortical arealization, artificial networks of a range of sizes were trained and tested. Feedfoward networks consisted of two hidden layers, with the knockout nodes residing at the first, and larger networks were constructed by adding hidden nodes to both hidden layers in equal numbers. Between five and ten networks, each with different random initial weights and training sequences, were evaluated for each size. For each topology, reconstruction errors reduced exponentially as network size increased, reaching a plateau for networks approaching 30 nodes in size (Fig. 3.3).

Where performance plateaued, networks were consistently able to recreate up to fourteen spatially distinct and correctly localized fields, of a possible sixteen (four fields in each of four conditions). Some combination of the substantially reduced A1, V1, and S1 fields were often missing from the reconstructed Coup-tf1 knockout patterns, owing to the under-representation of these fields in the training set. We therefore identified the network with the highest reconstruction error that could reproduce fourteen fields, and used its reconstruction error (0.011 error per pixel per output node), post hoc, to determine whether reconstructions were 'good' or not.

Against this criterion, reconstructions of the observed arealization patterns obtained via either network topology were good when networks consisted of 17 nodes or more. Feedforward networks required slightly fewer nodes, ~ 17, than recurrent networks, ~ 19. Importantly, both yielded estimates of network complexity, in terms of the required number of thresholding operations, that are on the same order as the 20-40 genes currently implicated in neocortical arealization (see Rakic et al. (2009)). We might have alternatively estimated network complexity based on the number of

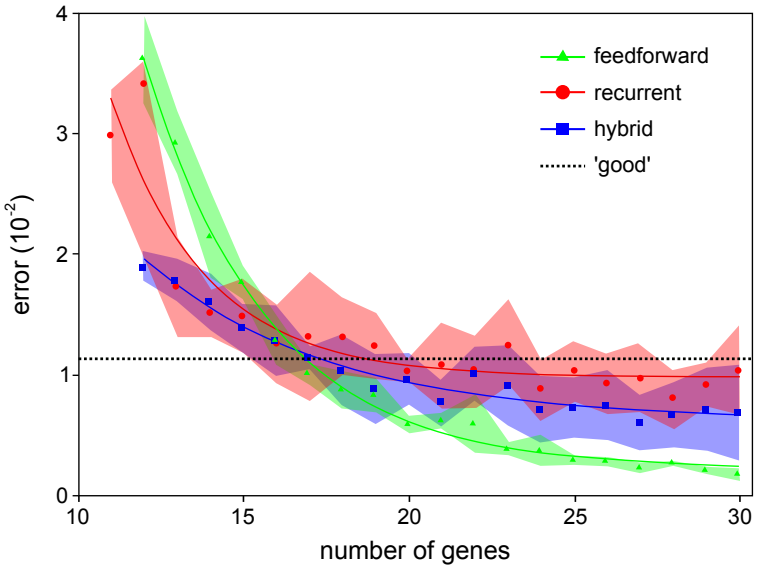


Figure 3.3: *Estimating the complexity of gene-interaction networks.* Feedforward, fully recurrent, and hybrid networks of different size were trained to simultaneously reconstruct the wildtype and knockout patterns of cortical arealization. All networks consisted of two input nodes, three knockout nodes, and four output nodes. Feedforward nets consisted of two equally sized layers of hidden nodes, with knockout nodes residing at the first layer. Hybrid networks consisted of an intermediate layer containing the knockout nodes and two additional hidden nodes to which the inputs projected, and a subsequent pool of fully inter-connected nodes which included the four outputs. In each case, as nodes were added to create larger networks, the reconstruction error reduced exponentially (as depicted by solid lines showing exponential fits). All three network topologies yielded good performance when networks contained approximately $\sim$ 17–19 nodes, consistent with estimates of the size of the gene interaction network for neocortical arealization.

nodes at which the reconstruction error plateaued, which would seem to favour the feedforward topology, however inspection of the reconstructions here reveals that the performance gains were marginal, overfitting the data at the resolution of individual image pixels rather than capturing any additional spatial logic.

3.2.2 Compatibility of learnt artificial networks with biological network structure

Feedforward and fully recurrent networks of similar complexity can be configured to simultaneously recreate cortical arealization patterns observed in wildtype and knockout mice. But how do these compare as models of the biological network? We can address this question by examining the structure of the networks, paying particular attention to the patterns of expression produced by nodes identifiable with specific genes.

In the feedforward networks, knockout nodes were included in the first layer, in part to reflect the fact that patterns of expression of Coup-tf1, Emx2, and Pax6 are apparent in embryonic mouse brains several days earlier than are the patterns of expression of area markers such as Id2 and RZRβ. The responses of these nodes are given by a rotation of the coordinates of the tissue through the learnt connections to the x and y input nodes, by a translation through the learnt bias weight of the node, and by a (sigmoidal) compression along this axis. Thus, responses of nodes in the first layer appear as tissue-wide monotonic oriented gradients, resembling those observed in the developing cortex. As such, the orientation of the expression gradients of the knockout nodes can be computed directly and compared to the known orientation of the corresponding gene expression. Fig. 3.4 shows the distribution of orientations ob-

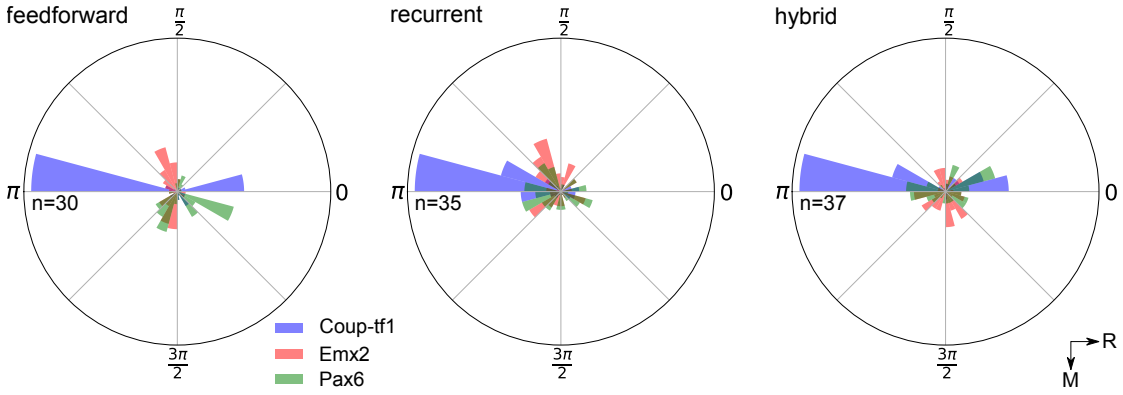


Figure 3.4: *Comparing artificial and biological network organization.* Polar histograms show the distribution of the orientations of the expression gradients produced by the three knockout nodes in all good networks of a given topology. This analysis shows that the orientation of Coup-tf1 expression gradients was biologically consistent (between 165° and 180°; from low rostromedial expression to high caudolateral expression) in the majority of successful networks, while Emx2 and Pax6 expression was not sufficiently constrained by the training data. Sample sizes of good feedforward, recurrent, and hybrid networks were 72, 103, and 97 networks, respectively. Histograms were individually normalized, so the number of samples in the bin of maximal size, n, is indicated in each plot.

tained from all networks that produced good reconstructions. Interestingly, only the expression pattern of the Coup-tf1 node was consistently orientated, overwhelmingly at around 170°, corresponding to the rostro-medial (low) to caudo-lateral (high) expression of this gene across the ventricular zone. Expression gradients of the other knockouts varied more uniformly amongst the candidate networks, suggesting that the logic captured by the training patterns is not prescriptive at the level of constraining Emx2 and Pax6 expression to be graded caudomedially or rostrolaterally, respectively.

It was conceivable a priori that the expression patterns of knockout nodes in fully recurrent networks might also have produced simple oriented gradients, but they did

not. Instead, settled patterns of expression produced by all network nodes were more complex, with distinct regions of high and low expression in each, usually against a background of overall graded expression. We therefore determined the general direction in which this background expression was graded in the knock-out nodes, by comparing them to templates representing linear gradients through the range of orientations, and inspecting the distribution of the optimally fitting template orientations. This analysis revealed similar trends; overall expression patterns of Coup-tf1 were consistently oriented as in biological nets, while Emx2 and Pax6 patterns were not sufficiently constrained by the training data.

3.2.3 Spatial temporal pattern formation in a hybrid network architecture

Although only feedforward networks produced realistic expression patterns in the knockout nodes, real networks are known to be highly recurrently connected. Thus we explore here a hybrid network architecture consisting of a feedforward layer that projects to a 'pool' of recurrently interconnected nodes. With this architecture, superior reconstructions were obtained from networks in which the feedforward layer contained two hidden nodes in addition to the three knockout nodes, to reduce the information bottleneck projecting to the recurrent pool (see Supplementary materials S1). Otherwise, hidden nodes were added to the recurrent pool only. For an equivalent number of nodes, reconstructions obtained with this hybrid architecture were of a similar quality to those produced by the feedforward and recurrent networks (Fig. 3.2). Good reconstructions were obtained from hybrid networks of equivalent complexity to good feedforward networks, $\sim$ 17–18 nodes (Fig. 3.3). And expres-

sion patterns in the knockout nodes were monotonically graded, with biologically consistent orientations for reconstructed Coup-tf1 expression only (Fig. 3.4).

As well as inheriting the advantages of the feedforward architecture, the recurrent processing loops of the hybrid architecture give rise to spatial patterns of expression that emerge and become dynamically stable over time (Fig. 3.5). Interestingly, while the patterns produced by the knockout nodes are fixed, and the output nodes generate distinct cortical field shapes that emerge gradually as the network activity settles, the spatial temporal expression profiles of the hidden nodes in the recurrent pool are more complex, with tissue-wide gradient-like patterns that appear early giving way to distinct patterns of expression that become polarised in several different locations and clearly come to overlap with multiple cortical fields. We note that qualitatively similar spatial-temporal profiles have been observed experimentally Dye et al. (2010a,b), unfolding over a period of embryonic and early postnatal development that lasts for several days.

3.3 Discussion

A key challenge to reverse-engineering gene interaction networks is that very many possible network structures may be consistent with the observed effects of manipulating individual genes Bhalla (2003); Guet et al. (2002); Hasty et al. (2001). One way around this problem is to use learning algorithms developed for artificial neural networks to generate candidate models automatically, by training artificial gene networks on data derived from multiple experiments. Here we developed such an approach, using an error backpropagation algorithm suitable for training networks of

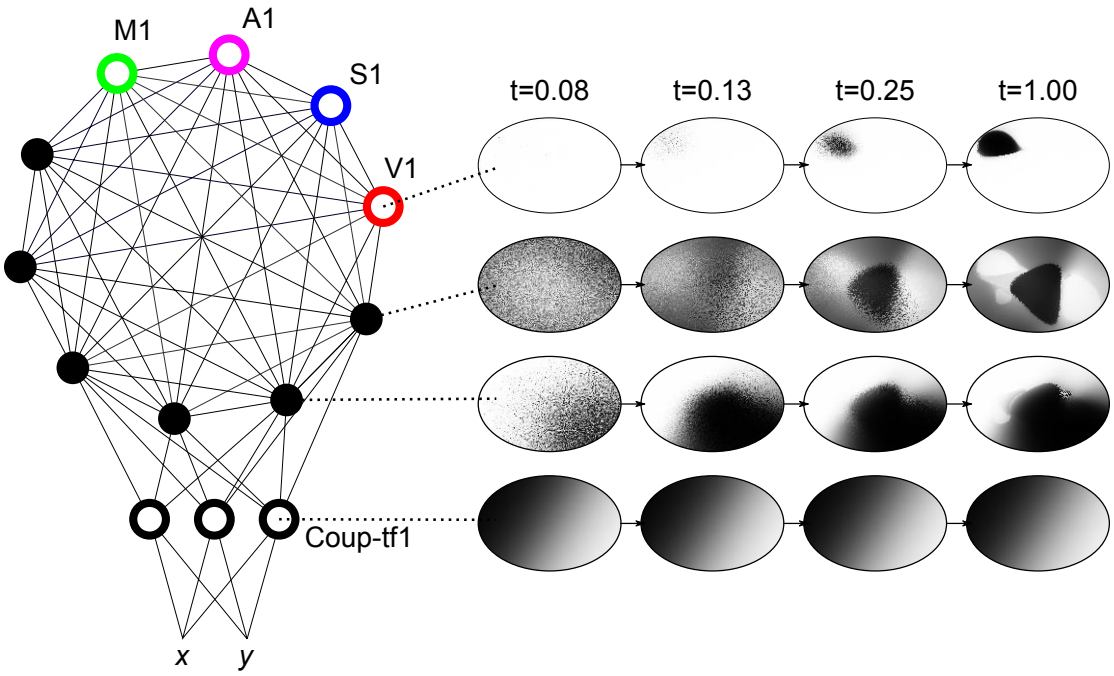


Figure 3.5: *Spatial temporal pattern formation in a hybrid network for cortical arealization.* The hybrid network architecture is depicted on the left, with input nodes x and y projecting to a hidden layer containing Coup-tf1, Emx2, and Pax6 knockout nodes, which in turn project to a pool of recurrently connected nodes that include the area markers for V1, A1, S1, and M1. After training, the time evolution of the responses of four nodes, to presentation of co-ordinate values to the two input nodes, is shown on the right. Time units t are scaled so that $t = 1$ depicts the settled response. Knockout nodes (bottom) have stable patterns, and cortical field shapes appear gradually in the responses of the output nodes (top), whereas more complex spatial-temporal patterns emerge in the hidden nodes of the recurrent pool. The network from which the responses shown were derived contained fifteen additional nodes (twelve in the recurrent pool) that are omitted for clarity.

arbitrary topology to reproduce the variation in cortical field shapes observed across a range of knockout experiments. Analysis of a large sample of candidate networks generated in this way allowed us to estimate some basic features of the design of the gene network for neocortical arealization, including its logical complexity and likely topological structure. This investigation therefore demonstrates, in more general terms, how learning algorithms can be used to reverse-engineer the gene network interactions that produce complex 2D patterns of expression, in line with the basic mechanisms assumed by the positional information model (Wolpert, 1969, 1996).

When the goal of training artificial nets is to classify data between groups, standard practice is to train using large datasets comprising many instances from each group, to validate by training and testing on different subsets of the full dataset etc.. But as the goal of training networks here was to very accurately re-represent the very specific shape information that pertains to a very small number of cortical fields, applying these standard evaluation criteria (promoting generalizability, avoiding overfitting etc.), would have been meaningless. Instead, candidate nets were evaluated in terms of their reconstruction errors (networks producing better fits are better candidates), in terms of the biological plausibility of their reconstructions at the resolution of individual nodes/genes (networks of nodes that each respond more realistically are better candidates), and in terms of their overall complexity (all else being equal, a net with fewer nodes/connections is a better candidate).

Against these criteria, our analysis of a large sample of candidate networks suggests that $\sim$ 17–19 non-linear thresholding operations must interact to specify the shapes of the primary visual, somatosensory, auditory, and motor fields in mice, in such a way that removing operations attributable to Coup-tf1, Emx2, and Pax6 results in the specific patterns of distortions observed in corresponding knockout ex-

periments. In addition, our analysis favours a 'hybrid' network architecture, where these transcription factors reside in a feed-forward layer that projects to a pool of recurrently interconnected genes, whose interactions generate spatial-temporal patterns of expression mirroring those observed during late embryonic development.

Let us now consider how techniques from artificial neural networks research may be useful for reverse-engineering gene networks more broadly.

It is first important to consider that the individual 'nodes' in biological networks may be capable of specifying more complex mappings than can be achieved by individual nodes in standard network models, as represented here using sigmoidal threshold operators. Minsky and Papert (1969) famously demonstrated the limits of perceptrons by way of the exclusive-or (XOR) problem – simple threshold operators cannot signal when only one of two input values is high. In similar terms, when we look to equate nodes with genes, our approach implicitly assumes that not all possible logical operations may be carried out by individual genes. However, a model of gene interactions formulated at the level of protein-binding equated the individual nodes of a comparable (probabilistic) neural network model with the individual binding sites in the cis-regulatory region (Buchler et al., 2003), and showed how the XOR function of two gene expression levels can in fact be computed by the molecular machinery of what would here correspond to a single network node (see also Saghatelian et al. (2003); Ashkenasy and Ghadiri (2004); Baron et al. (2006)). Thus it is important to acknowledge that in some cases individual gene interactions might be more appropriately represented by a small number of interconnected nodes. A weaker interpretation of our analysis is therefore that the number of artificial nodes required for good pattern reconstruction (~ 17) represents an upper bound on the estimate of the network complexity.

Nevertheless, it is interesting that our analysis yielded an estimate of the complexity of the arealization network that is broadly consistent with (and certainly does not exceed) the number of genes currently thought to be involved in cortical arealization. One interpretation, perhaps the most parsimonious, is that individual interactions between the genes involved in neocortical patterning are relatively simple, in the sense that the logical OR and AND operators, which can be specified by a single decision boundary, are simpler than operators such as XOR, which require a configuration of multiple decision boundaries to realise. An alternative interpretation is that the number of genes involved may be less than estimated using our approach, with biological networks exploiting redundancy, or with many of the genes currently implicated playing a more substantial role further downstream of those responsible for the gross shaping of the primary fields. Note that in Figures 3.6 and 3.7 we report on simulations in which individual genes were represented by small clusters of nodes, which suggest that while the computational complexity (i.e., the total number of required threshold operators) is unaffected, cortical patterning may be specified by as few as nine genes.

A final consideration, as explained in the original article by Bray (1995), is that the transfer function of nodes in biological nets may in some cases be more accurately described as hyperbolic rather than sigmoidal (see also Stirling and Laughlin (2015)). Indeed, biological nets may constitute a mixture of hyperbolic and sigmoidal operations. Thus it is important to note that the approach developed here is easily extensible. As Pineda's backpropagation algorithm can be expressed as a local learning rule, and as the hyperbolic function, like the sigmoid, is differentiable, a similar methodology could in principle be formulated for networks containing nodes of either type.

We, like others, advocate an 'ensemble approach' to understanding gene regulatory networks, e.g., (Bray, 2003; Kauffman, 2004) – somewhere between studying the details of small circuits in isolation from external systems-level effects, and using machine learning techniques to process large systems-level datasets obtained via high-throughput gene assays. The study of abstract models like Boolean networks can help identify regions in possibility space where gene networks may produce observed dynamics, as has already been successful in the context of cortical arealization (Giacomantonio and Goodhill, 2010; Wilson et al., 2019). But our work here aims to go further, by using a modelling approach that is more naturally suited to the study of the rich continuous-valued two-dimensional patterns that gene nets can paint onto tissues like the developing neocortex. Here we analysed summary information depicting knockout arealization patterns. But the approach is well suited to analyse primary data, when these become available in a common coordinate system that permits patterns obtained from different experiments to be compared directly.

Neural networks are currently being used so effectively in machine learning applications, as black boxes in which to discover statistical structure embedded in large datasets, that it is easy to overlook their potential for representing biological processes more literally, as we have done here, and as was fashionable in the era before big data. Neural nets need not be black boxes – identifying specific genes with individual operators in artificial networks may help us unlock valuable information that may otherwise remain hidden behind the complex dynamical patterns produced by biological networks.

3.4 Methods

According to the recurrent backpropagation algorithm, external inputs are presented via an input vector **I**, which has components that are zero for all but the 'input nodes'. Target responses are presented via output vector **J**, with components that are $\mathbf{J}_i = \text{target}_i - x_i$ for the 'output nodes', and zero for all others. In response to a given input, activity, x, is propagated through the network, with w_{ij} the connection weight *from* unit j to unit i, by iteratively computing $u_i = \sum_j w_{ij} x_j$, applying the logistic function $\sigma(u) = (1 + e^{-u})^{-1}$, and settling $\frac{dx_i}{dt} = \sigma(u_i) - x_i + \mathbf{I}_i$. In practical terms, once the network activity has had time to settle it is useful to work with the average error signal computed from the responses recorded over several recent iterations. The resulting error signal is propagated back through the network by iterating $\frac{dy_i}{dt} = \sum_j w_{ji} y_j \sigma'(u_j) - y_i + J_i$, where $\sigma'(y) = \sigma(y)(1 - \sigma(y))$ is the derivative of the logistic function. Finally the weights can be adjusted to reduce the reconstruction error with $\frac{dw_{ij}}{dt} = \eta x_j y_i \sigma'(u_i)$, where η is a time constant parameter. The reconstruction error is error $= \frac{1}{\mathrm{n}} \sum_{\mathrm{i}}^{\mathrm{n}} \mathrm{J_i}^2$.

According to the analysis of Pineda (1987), the dynamics will converge for most patterns of weights, however convergence is not guaranteed, so it is useful to either perturb the weights by a small random amount whenever a maximum number of settling steps is exceeded, or to time-average the final steps, which is the approach used for the analyses reported. We set $\eta = 0.05$.

The four images used for training networks were digitally traced, and each pixel p was assigned a coordinate, x_p, y_p, between 0 and 1, based on its location in the image. Images were sampled at a resolution of 100 by 150 pixels. Each pixel was labelled as "M1", "S1", "V1", "A1" or "None". To each label we associate a target pattern,

defined with respect to four output nodes, so that the target pattern corresponding to the label "M1" was target $= (1, 0, 0, 0)$, the target corresponding to "S1" was $(0, 1, 0, 0)$, the target for "V1" was $(0, 0, 1, 0)$, that for "A1" was $(0, 0, 0, 1)$, and that for "None" was $(0, 0, 0, 0)$. Generating a single training pattern for the network corresponds to choosing a single pixel from one of the images, using its coordinate to specify the input values $I_1 = p_x$ and $I_2 = p_y$ and using its cortical area identity to specify the target vector for the four output nodes.

In addition, three hidden nodes are identified with transcription factors Coup-tf1, Emx2, and Pax6. During the training and testing of a network, the response of one of these hidden nodes is overwritten on each timestep to have an activation of $x_i = 0$ whenever the area identity of the target output pattern is chosen to be from the corresponding knockout image. Training patterns were drawn randomly from the wildtype and the three knockout conditions with equal probability. This artificial knockout technique was developed to ensure that networks learn a pattern of interactions that is consistent with the known effects of knocking out a particular transcription factor, and to identify these network nodes with specific genes in order that inferences about their role in biological networks can be derived from inspection of their connection weights.

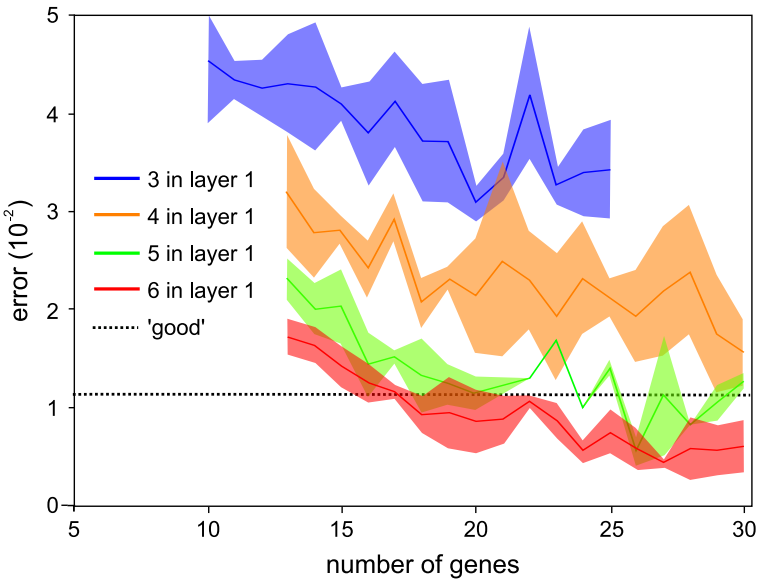


Figure 3.6: *Experiments with the configuration of the hybrid network architecture.* Hybrid networks consisted of a feed-forward layer (layer 1) containing the knockout nodes and a subsequent pool of recurrently connected nodes including the output nodes. Reconstruction improves as the number of nodes in layer 1 increases, though allocating six nodes to layer 1 constitutes only a marginal improvement over allocating five nodes to layer 1.

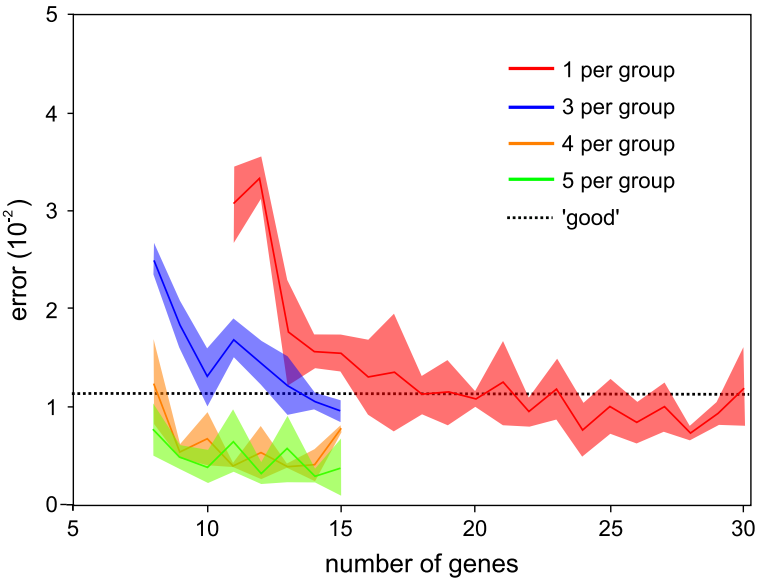


Figure 3.7: *Representing genes as clusters of nodes.* Each gene was represented by a small group of nodes, one of which received input only from all others in its group, and projected its output to all other groups. The '1 per group' simulations were equivalent to the fully recurrent networks reported in the paper. For a simulation of size 10 genes, the '3 per group', '4 per group', and '5 per group' networks comprised 30, 40, and 50 nodes respectively. Grouping nodes in this way represents an assumption that individual genes can compute more complex functions of their inputs, and as discussed in the main paper can here be seen to yield a less conservative estimate on the size of the gene network for cortical arealization.

Chapter 4

The effect of different mutation types on evolvability

4.1 Introduction

In Chapter 2 we looked at the interplay of evolution and development of spatial patterns with a simple abstracted model. Chapter 3 then delved into the creation of a more realistic biological model which can create accurate spatial patterns mimicking the development of, for example, neocortical areas. Now we would like to tie these two aspects together and use the cortical pattern model of chapter 3 to explore in what ways a developmental process is important for the ability of a species to evolve. Evolution involves both mutations at the level of DNA and then selection of the fittest at the level of the phenotype, however it is development which determines the latter from the former. The neural network model of the previous chapter portrays the development of gene expression patterns (a phenotype). The model describes the

interactions between genes without saying how mutations to those genes affect the network. The necessary next step is to address how DNA mutations affect the gene, the network, and how this relates to our model representation of it. If we can use the model to accurately determine how mutations affect the phenotype then we have a method for investigating the interaction of evolution and development, the goal of this thesis set out in Chapter 1. In particular I will address two questions here. Do certain types of mutation create differing phenotypic distributions? Do certain types of mutation affect the ability to evolve?

Firstly, lets overview what has been said historically in evolutionary theory about the role of development and understand the motivation for asking these questions. Theories of evolution have progressed in three broad stages, (i) Darwin's initial conceptualisation, (ii) incorporation of genetics and statistics (the Modern Synthesis), and (iii) the contemporary stage including epigenetics, genomics and developmental biology (the Extended Synthesis).

Darwinism in brief is the theory that species diversify, from a common ancestor, by variations in the population being passed on selectively. The theory is concisely stated in the title of Darwin and Wallace's groundbreaking paper, 'On the tendency of species to form varieties; and on the perpetuation of varieties and species by natural means of selection." (Darwin and Wallace, 1858). It was mostly accepted at the time although the controversial aspect was that the selection of useful variations is done by 'Nature', what Spencer would later call "survival of the fittest". We might now take it for granted, but at the time it was unacceptable that the selection of individuals according to small differences, in a natural unguided way, could lead to qualitative differences such as the ability of flight. Even Darwin himself assumed that natural selection couldn't be the exclusive cause of such adaptations. Of importance

to our investigation, the theory assumes that populations have plenty of variety for nature to work with, yet the origin of the variety was unknown and didn't need to be known for the theory's success. It was readily accepted that variation existed; what was needed was an explanation of how beneficial variety is perpetuated. However, from the perspective of this work, the question of how phenotypic variation arises from random mutation does require explanation, as it relates to the strength or rate of evolution.

As well as not explaining the origin of variation, Darwinism did not account for inheritance of traits. The 'perpetuation of varieties' was not well understood, as Mendel's theory of heredity was unknown to Darwin and Wallace. Mendel's discovery, that variations in pea plants were inherited by combining genetic material of the parent plants in a discrete way, was largely ignored for 35 years. Then when Bateson rediscovered Mendel in 1901 it was controversial and seemed to deal a further blow to the already maligned Darwinian theory (Keynes and Cox, 2008). The problem was that Mendel's discrete heredity seems to go against the gradual change proposed by Darwin. Remarkably, while an explanation of heredity provides an improvement to Darwinism, at the time it was thought to be its refutation. Thankfully in 1918 Fisher showed that discrete Mendelian traits, when affected by several genes, could lead to a normal distribution of phenotypes (Fisher, 1919) . This 'bell-curve' of phenotypes gives natural selection a pool of variability which it can work with in the gradual way expressed by Darwinism. Further work by Fisher, Haldane and Wright developed the rigorous mathematics of population genetics at the heart of modern evolutionary theory (Haldane, 1932; Fisher, 1930; Wright, 1932). By incorporating this work with all classical disciplines of evolution the Modern Synthesis (coined by Huxley (1942)) was fully developed by 1950.

The Modern Synthesis has been the cornerstone of evolutionary thought for 80 years. It has been useful for understanding countless questions, while taking as given that biology provides a distribution of phenotypes that natural selection can work with. Once DNA was better understood (Watson and Crick, 1953), which gives a source of variation other than recombination of alleles, the Modern Synthesis was unaffected. The reason is that the distributions of phenotypes seen empirically, regardless of their cause, are found to comply with the assumptions of the Modern Synthesis. It's methods can safely be used without asking how phenotypic variability arises, be it from DNA mutation or allele recombination. However, we are interested in the 'how' and 'why', because we assume that it effects the rate of evolution or evolvability. If the rate of evolution is effected by the distribution of phenotypes, then how that distribution is determined from DNA mutations is important to understand.

The need for the Modern Synthesis to incorporate new aspects such as Evolutionary Development is largely undisputed. A new framework known as the Extended Synthesis is now being developed which aims to understand evolutionary questions previously left out, such as epigenetics, evo-devo, multi-level selection and evolvability (Pigliucci and Muller, 2010). Here I would like to argue that the source of variation is indeed relevant to understanding evolution. We now know that the sources of variation are mutation and developmental processes. Their effect on evolvability is the focus of this study.

The Modern Synthesis idea that development is not required in the explanation of evolution relies on three fundamental assumptions. These are that: (1)Additive genetic variation is normally available for most traits. (2)The relationship between the phenotype and the genotype is simple. (3)Phenotypic variation is gradual and normally distributed. (Salazar-Ciudad, 2006) The first and third assumptions are

fairly uncontroversial. We see empirically the normal distribution of phenotypes all the time. We also know that almost any additive genetic variation is feasible, as will be discussed in detail later in this chapter. But the second assumption is where development plays its role. With all the complexities of developing a phenotype from genotype there is in fact no reason to assume that the mapping is ever simple. There are plenty of mechanisms through which tiny changes to the genome can have catastrophic effects on the phenotype.

So while we may observe the neat bell curve of phenotypes that the foundations of evolutionary theory assume, it is rarely a sufficient explanation that these arise simply from random mutations. Developmental processes play a key role in determining the distribution of phenotypes.

Some types of mutation cause more severe disruptions to the phenotype, in which case there may be mechanisms within the replicative processes which prevent such mutations. A remarkable example is that of the experiments with the Hox gene in the fruit fly, where a single mutation was found to cause legs to grow where the antennae should be (see Figure 4.1)(Lewis et al., 1980). We can assume that deleterious mutations do sometimes occur in the wild, so long as there is a sufficient probability of beneficial mutations to allow the species to evolve. Our question is therefore how developmental dynamics can help to tip the balance in favour of beneficial mutations, and thus improve evolvability.

The neural network model of the previous chapter, showing how spatial patterns arise through the interactions of genes, provides a useful framework. Within our model it is possible to trial a complete set of mutations and observe the distribution of phenotypes developed. By using an evolutionary algorithm we can then investigate

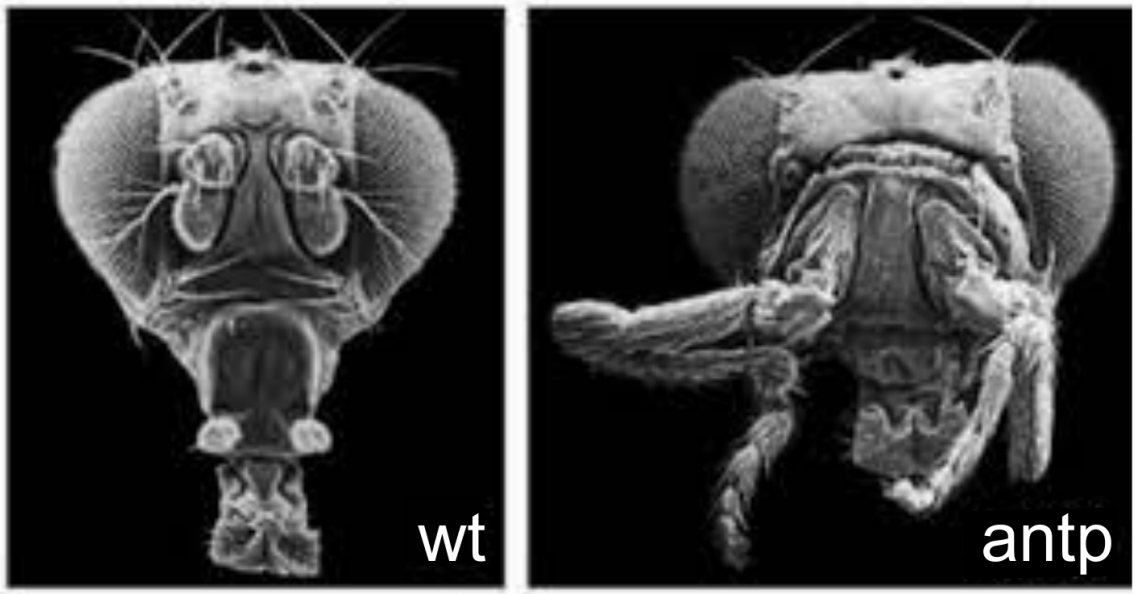


Figure 4.1: A single mutation to the Hox gene causes the antennae of the fruit fly to be replaced by legs. (Lewis et al., 1980)

how the mutations and their phenotype distributions affect evolvability. However, in order to do this reliably we must first answer the question of how real biological mutations can be emulated in our gene network model.

4.2 Effects of Mutation

We can picture the effects of mutations at four levels of emergence, each affecting the next. Most straightforwardly a mutation affects the structure of DNA, changing the sequence of nucleotide bases. This new sequence affects the coding and regulation of a single gene. The properties of that gene and the protein it produces cause changes to the function of the gene network. Finally, changes to a gene network can affect the phenotype of the organism. Our developmental model addresses the final part but for

the earlier stages there are still many open questions. There are many complexities related to the biochemistry of a single mutation, which require multiple disciplines to understand. It is like investigating the effect of removing a single transistor in a computer chip on a computer program, requiring knowledge of circuitry, hardware, operating system and code. At each level there are new complexities.

In order to approach this cascade of dependencies I will take each level and sketch out the range of possible effects. From there we can speculate on the mapping of changes from one level to the next. Figure 4.3 shows a summary of the possible changes that can occur at each level.

The most well understood effects of mutation happen at the level of DNA as the causation is direct. Here we can consider two broad types of mutation - chromosomal mutations affect whole sequences of bases in the chromosome, whereas point mutations affect single bases. Chomosomal mutations are more likely to have catastrophic effects. For example, disorders such as Down's syndrome are caused by such dramatic changes to a human's chromosome. Possible mutations of this type include entire sections of the chromosome being deleted, duplicated or inverted. It is also possible for one section of chromosome to be moved and inserted into another (insertion), or for it to be swapped with another (translocation). Clearly such disruptions to the DNA could have an enormous range of effects at the higher levels of genes, networks and phenotypes. This actually makes our job somewhat easier because almost any change we can imagine at the higher levels could feasibly happen. As such mutations often cause deleterious effects, their rate is often reduced, eg. via DNA repair. Effecting the rate of mutations is one way systems can effect their own evolvability.

Compared to chromosomal mutations, point mutations are much more common

and less severe. The effects on DNA are fairly straightforward; inserting, deleting or substituting a single nucleotide base. The effects on the gene are not so simple and again the range of consequences is fairly comprehensive. Although I have mentioned only three possible DNA changes here, consider also that a change in the coding region has very different effects to a change in the non-coding region. Point mutations in the coding region can be classified as silent, missense and nonsense mutations. Silent mutations have no effect as the same amino acid is coded for via a slightly different triple of bases. Missense mutations can have a range of effects as they cause a different amino acid to be coded for. Finally, nonsense mutations have very serious effects as they alter the transcription of the entire sequence by creating a premature stop codon. In the coding region we could also consider frameshift mutations. As the DNA code is read in codons of triplets, inserting a single base can cause a different interpretation of every subsequent codon by shifting the reading frame. This can completely change the protein produced and be disasterous.

The vast majority of DNA is of the non-coding type, with long sequences of bases simply creating space between binding sites for transcription factors. Mutations in this region are far more likely and will cause more gradual changes to the gene. Thus another example of how developmental processes can make the system more evolvable. The possible DNA changes in the non-coding region are the same as in the coding region; insertion, deletion and substitution. In this case however, they affect the regulation of the gene rather than the protein that it codes for. To see how these mutations affect regulation lets now turn to the gene and consider all alterations at that level. From there a mapping from DNA mutation to gene variation should become clear.

A single gene is the sequence of DNA which codes for a single protein, but also

regulates the transcription (and thus expression) for that protein. The bases of DNA, as well as embodying the code for construction of amino acids, have structural properties that allow proteins to bind. A different sequence will bind a different protein, or the same protein but with a different strength. The bound proteins then have multiple effects such as interacting and binding with each other, changing the 3D structure of the DNA (looping), and binding to other transcription machinery such as RNA Polymerase. The overall effect is that the bound proteins regulate the transcription of the coding-region, and are therefore called Transcription Factors.

Thus the string of DNA bases is in fact a series of binding sites, the spacing of which has functional importance, alongside a coding region where the gene is transcribed. Each gene can have several sections of binding sites which regulate transcription independently, which makes regulation modular (see Chapter 1 for further detail). The importance of this modular system to our description of the gene is that changes to the gene can effect either binding sites, whole modules, or the coding region itself.

We can envision the following possible changes concerning binding sites:

- a new site is created
- an old site is lost
- the number of sites changes
- the spacing between sites changes (or they overlap)
- the ordering changes
- the strength of binding changes

There are more nucleotide bases between binding sites than there are at the sites themselves. Therefore we can assume that the most common change resulting from point insertions or deletions would be a change in the binding site spacing. If a nucleotide is substituted within a binding site it will almost always change the strength of binding, rather than destroy the site completely. Experiments with Drosophila have shown the binding strength change for every possible point mutation within a particular site (Jung et al., 2018). So most point mutations in the non-coding region will lead to a change in spacing or strength of binding sites (or to no change at all).

This is already enough to account for a computational ability comparable to a Boltzmann Machine (Buchler et al., 2003). However, the more substantial changes such as new or lost sites and ordering changes are also possible. Their effect at the gene network level is not necessarily severe as I will later explain. Possible changes that are likely to be severe include the creation and deletion of entire modules and alterations to the coding sequence as previously mentioned. For a review of changes at the CRM level and their evolutionary consequences see Peter and Davidson (2011).

The next level up is that of the Gene Network, which in Chapter 3 we modelled in terms of a recurrent Artificial Neural Network (ANN). Again it is straightforward to envision all possible changes to the network, but in this case the mapping from the previous level is more speculative. The network as we model it is a weighted directed graph. At this level of description possible alterations are to:

- add or remove nodes
- add or remove links connecting nodes
- change the weight of a link

- alter the boolean logic (a whole set of weights)
- change a node threshold (or bias weight)

Considering that the network weights can be changed by any size and in any combination gives an enormous range of effects. But the question remains what subset of these changes are analogous to changes at the lower level of the gene? This is an extremely difficult question in the case of the ANN, and is in fact the motivation for constructing a new model network in the upcoming chapter. For now we will limit ourselves to considering whether binding site changes at the gene level will lead to 'small' or 'large' weight changes at the network level.

For all of the possible changes to the binding sites of a gene, I'd like to show that many are analogous to changing a weight of the model network. The key to this is understanding the biochemical behaviour of an individual binding site. The details of this are in Chapter 1. Essentially, the probability of a site being bound is a function of the concentration of the ligand to be bound. Two parameters alter the features of this function: the dissociation constant (strength of binding), k, which determines the concentration where binding probability = 0.5 and the cooperativity, C, which determines the gradient at the central point (Stefan and Le Novère, 2013). Here is the function in full.

$$P(\text{binding}) = 1/(1 + (k/[L])^C) \tag{4.1}$$

Notice the similarity between this curve and the sigmoidal function used in the ANN in Figure 4.2. If we plot the response curve of a perceptron for a range of different weight values we can see how the k parameter has a similar effect. Most

point mutations in the sequence of a binding site will cause this effect of altering the k value, and therefore in our model a weight change. Another binding site change is the increase or decrease in the number of sites for a particular transcription factor. This too will result in a weight change (see bottom panel Figure 4.2). The reason for this is that if transcription requires that two sites are filled rather than one, the probability is $P(\text{binding})^2$. Multiplying this function by itself creates a very similar function but shifted to the right, much like increasing the k parameter by a small amount. Finally we can consider the most common kind of mutation, that which effects the spacing between sites. For this I refer to a proof in the supplement of Buchler et al. (2003) which relates the spacing between sites to the strength of the protein-protein interactions of TFs. They then show that the weights of an equivalent neural network (Boltzmann machine), are proportional to the logarithm of such an interaction strength.

At the phenotype level the potential variation is visible under microscope and well documented. Across mammals one can observe all sorts of changes to the cortical fields, but varied around the basic plan of their common ancestor (Krubitzer and Kaas, 2005). The cortical fields can vary in size, position, function and number. In this investigation we will observe the change in size of two model cortical fields after applying mutations at the network level. This can be quantified by assigning a fitness function to the phenotype according to field sizes.

To summarize, we have built up a picture of the complex cascade of effects that arise as a result of DNA mutation. The changes at each level cover a large range of possibilities. The mapping from one level to the next is non-linear and many-to-many. We hope to represent a substantial subset of these possible variations by modelling two kinds of mutation in the gene network. Small weight changes between

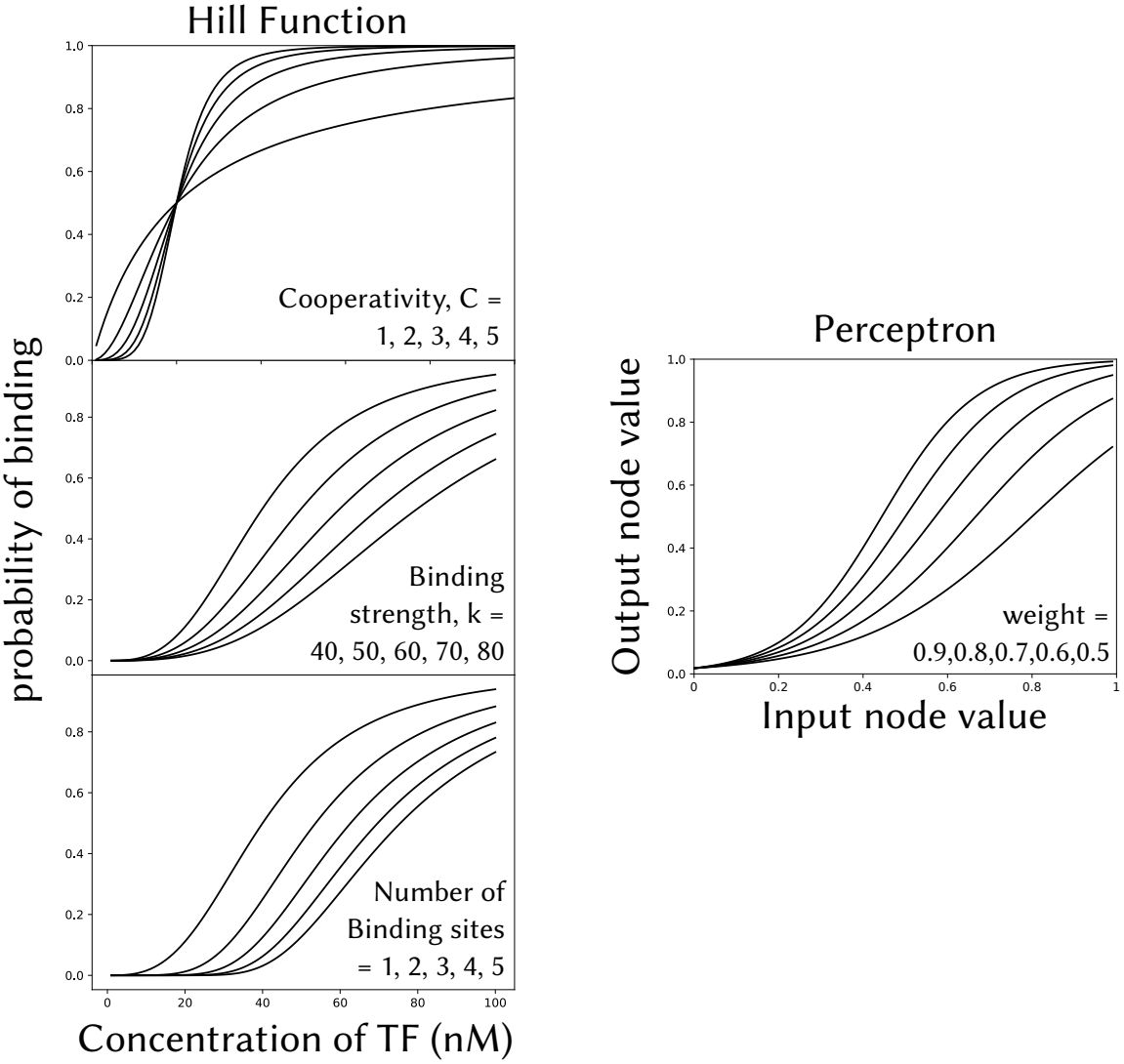


Figure 4.2: Comparison of Hill function (left) and perceptron response (right). The Hill function is $p(\text{binding}) = 1/(1 + (k/[\text{concentration}])^C)$ where k is 20 (top), varies (middle) or is 40 (bottom). C varies (top), or is 3 (middle and bottom). The perceptron is simply 3 inputs (including a bias) into an output nodes. One input is given a range of values and the output is plotted for weights between 0.9 and 0.5 (left to right). This shows that the binding-strength parameter, k, and the weight have a similar effect on the function. Although the similarity is less accurate, also changing the number of binding sites has a comparable effect to changing the perceptron weight. For completeness, in the perceptron the other node has value 0.3 and weight 1, the bias node has weight -0.7. The dot product of these values and weights are then squashed with the logistic function $1/(1 + e^{-10x})$.

nodes would arise due to point mutations in the CRM then affecting the binding site strength, spacing or number. The gain or loss of links between nodes would arise from more severe non-coding mutations, or more likely to the coding-region of the gene.

Two important ideas are worth restating and emphasising here. Firstly that the mutations possible in theory could lead to all sorts of effects and not just a normal distribution of phenotypes. Secondly, only a subset of the mutations theoretically possible can be allowed to occur in practice. Therefore there are features of the biological system that must constrain the likelihood of different mutations, improving evolvability. This investigation sets out to give more evidence in favour of these propositions by looking at the full range of phenotypes determined by different sets of mutations, and then measuring their rates of evolution.

4.3 Methods

Using the recurrent neural network model from the previous chapter we will train multiple networks to reproduce a simple cortical pattern. The 'starting' pattern, obtained through the usual network training, will be two circular fields. One could imagine they are a simplified version of the Visual and Somatosensory fields of the cortex. From there I will run two experiments. The first involves separately trialling every possible mutation and observing the distribution of phenotypes that emerge. The second runs evolutionary algorithms using varying conditions to obtain the 'final' pattern in which the left field has shrunk and the right has enlarged (see Figure 4.4). The independent variable in these evolutionary trials is the type of mutation which

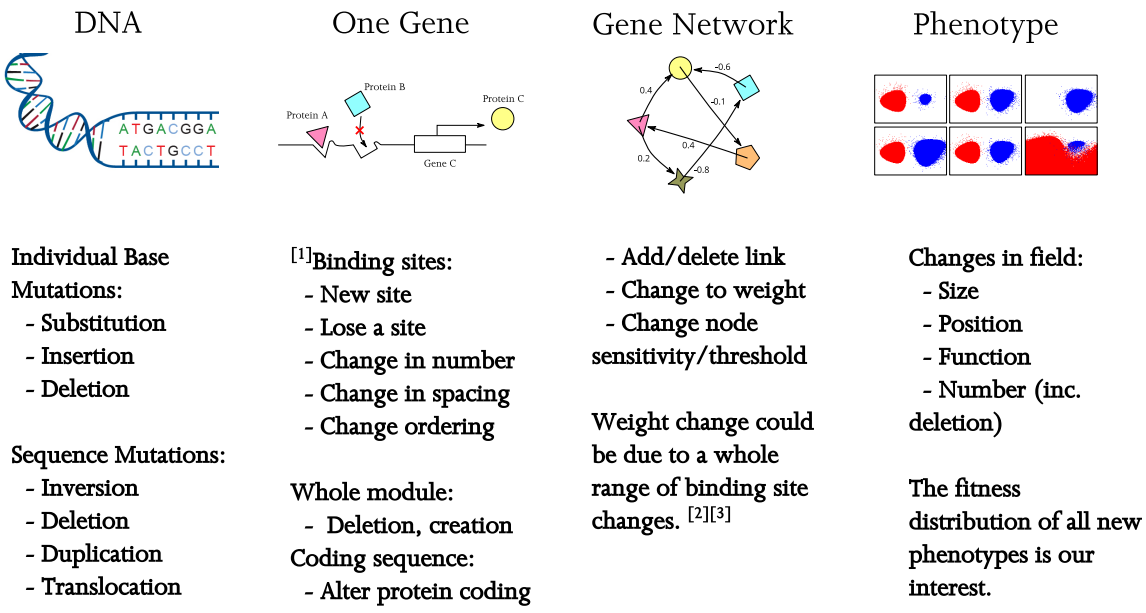


Figure 4.3: A summary of the possible changes that can occur at each level of development. Changes at the lower level cause changes at the higher level, but the mapping is many-to-many. In this chapter we focus on varying the links and weights at the network level, which we argue is possible due to a wide variety of lower changes. Then the fitness distribution and evolvability is what is observed. References: [1]Peter and Davidson (2011), [2]Buchler et al. (2003), [3]Wray et al. (2003)

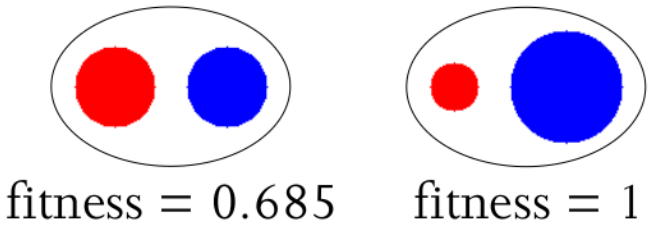


Figure 4.4: *Initial and final phenotypes.* The left phenotype is arrived at through backpropagation training the networks. The networks are then evolved to obtain the phenotype on the right.

is used. The dependent variable will be the observed rate of evolution.

The fitness function by which the produced patterns are judged and selected by the evolutionary algorithm is based on a normal distribution. Each domain (red,blue or white) has a target size. If that domain has the correct area it is given a fitness of 1.

$$\text{fitness} = \prod_{i}^{\text{red,blue,white}} (e^{-K(\frac{\text{area}_i - \text{target}_i}{\text{target}_i})^2}) \tag{4.2}$$

The fitness function is designed so that if the area is 0 or twice the target the fitness is 0.05. This is achieved by setting the constant, K, to be 3. The area is found by counting the number of pixels of that colour. Then the total fitness is found by multiplying the three domain fitnesses together. For the two experiments the target area needs to be slightly different. To measure the effect of a single mutation the target area should only change slightly. Whereas, when measuring the number of generations to evolve due to many mutations its helpful to aim for a bigger phenotypic change.

For each generation it is necessary to compute the fitness, which is the computationally intensive part of the algorithm. This could be done by observing the colour of all 15,000 pixels. In practice we took a random sample of 1000 pixels which was just as effective.

The first investigation will involve trialing every possible mutation just once and observing the distribution of phenotypes that emerges. The network used will be a 15 node network, with a range of connectivity (25%, 50% and 75%). The reason for being partially connected rather than fully recurrent is that I want to test for adding new links in as well as removing them. The network size of 15 was decided by measuring the lowest error attained in training for the initial phenotype, over a range of network sizes and learning rates. The error of producing the phenotype is much lower in networks of 14 nodes and above. The learning rate did not have a significant effect on the lowest error. With a 15-node network, with no self-connections, there are 15x14 = 210 possible connections. If the connectivity is 50% there are 105. For each one of these the weight can either be increased or decreased by a small amount. Therefore there are a total of 210 weight mutations that can be trialled for a half-connected network. Also, there are 210 possible link mutations as each possible connection can be added or removed depending on whether it already exists or not. When links are added their new weight is chosen randomly.

The weight mutations involve altering just a single weight by a small amount. We tried a range of values to change it by, ± 0.1, ± 0.3 and ± 0.5. These values were chosen because they create a change to the output comparable to those of the Hill function shown in Figure 4.2. The Hill function gives a way of estimating the effect of real mutations on the probability of binding. We'd like our perceptron model to emulate that as closely as possible.

For every single possible mutation the size of the red and blue fields will be measured, from this a fitness value of the phenotype can be calculated. The phenotype distributions of the weight mutations and link mutations can then be compared.

For the second investigation we will use these mutations in an evolutionary algorithm to change the sizes of the fields. The process for mutating the network involves going through each existent weight and with a certain probability changing the weight by 0.1 (0.3 or 0.5 in other trials). The weight is increased or decreased with equal chance. Similarly there is a probability that each link is either removed, created, or unchanged. It is possible to evolve the network with both of these mutations happening simultaneously with differing probabilities. This was done with a range of probabilities from 0 to 0.5 for both. In this case we are measuring the number of generations it takes to improve the fitness to a near optimal level (>0.95), over a range of mutation rates.

4.4 Results

Mutations don't necessarily lead to a simple normal distribution of phenotypes. Some types of mutation are much more likely to cause negative changes in fitness. Figure 4.5 shows the distribution of fitnesses of genotypes one mutation away from the initial network, for three different types of mutation. Link mutations, whether adding links or removing them, lead to a lower average fitness and a larger variance than weight mutations. The means and standard deviations of the distributions for weight mutations, adding links and removing links are (0.7,0.14), (0.61,0.23) and (0.39,0.33) respectively. Removing links also regularly leads to phenotypes that are

severly damaged, thus the mode of the distribution near zero. It should be noted that the fitness function creates an artificial cap at zero fitness. Therefore the large group of phenotypes approaching zero fitness are actually quite diverse, but all represent a significant departure from the target. If we take out these catastrophic phenotypes the mean fitness is still lower than weight mutations - (0.59,0.23). With its distribution shifted so significantly to lower fitnesses we should expect evolution to not proceed as efficiently through this mode of mutation.

What effect do other features of the network have on these distributions? We tested for sensitivity to the connectivity of the network and also the size of the weight mutation (see Figure 4.6). An ANOVA test revealed that the distributions could not be distinguished based on their number of connections. However, there is a small but statistically significant shift due to changing the size of weight mutation. The mean fitness of the distribution does slightly worsen as a result of increasing the magnitude of weight change. The change is so small (at worse 0.06) that it does not affect the rate of evolution as we shall see. So the previous result, that weight mutations produce a distribution of fitnesses which are more normal and have a much higher mean, compared to link mutations, holds true regardless of connectivity and size of weight change.

Now that we have established that some mutation types create different distributions of fitness we can ask how this effects the evolvability of the species. If a species were limited to one type of mutation or the other how would their rate of evolution fare? Figure 4.7 shows how many generations it takes to evolve to a 95% fitness. The phenotype begins once more as two fields of equal size, about 1900 cells each (circles of radius 25). The fitness is measured relative to a target size of 700 for red, 3800 for blue. This is a bigger change in phenotype than required for the

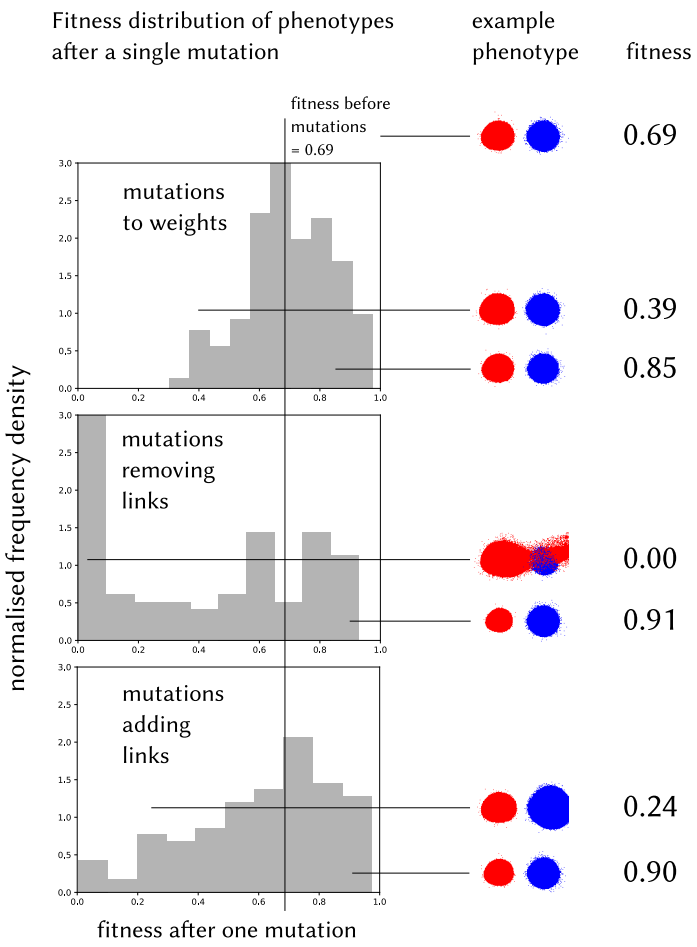


Figure 4.5: *Distribution of fitnesses after a single mutation.* By taking the same initial network and repeatedly applying every possible mutation a set of phenotypes is created. Their fitness is measured and plotted here as a histogram. The distributions from three mutation types are presented. A vertical line at 0.685 shows the fitness of the initial network. Several example phenotypes are shown on the right to give a sense of the effect of a single mutation. In this case the initial network is 50% connected and the size of weight mutations is 0.1. Clearly the biggest effect on the phenotype is a result of removing links, with a large peak in the distribution near 0 fitness. But adding links also results in a worse distribution than weight mutations, giving a distribution with larger variance and a higher proportion with worsened fitness. Mean and standard deviation for top and bottom charts are (0.7,0.14) and (0.61,0.23) respectively.

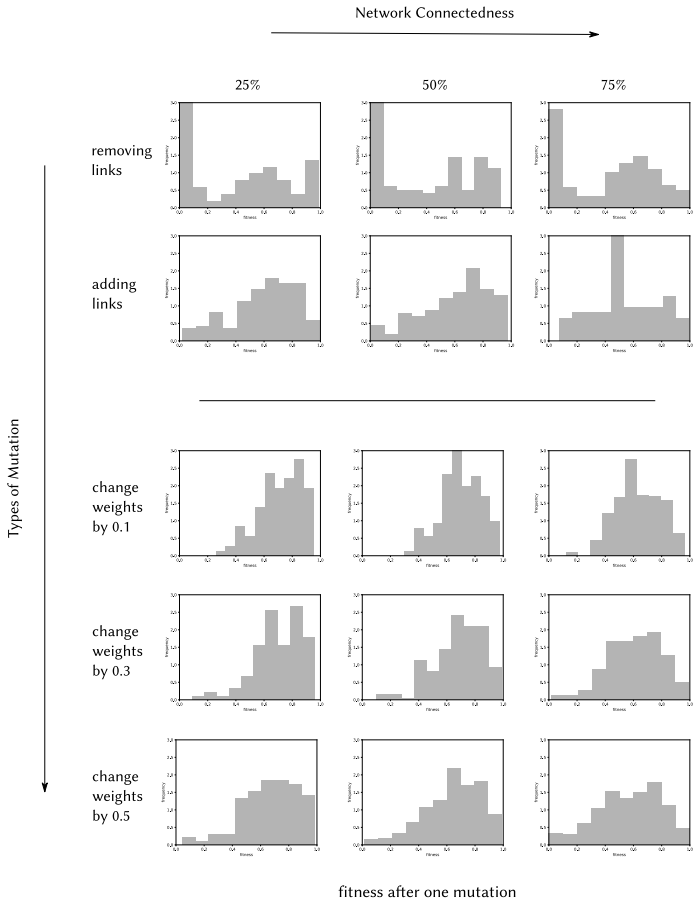


Figure 4.6: *Distributions mostly unaffected by connectedness or size of weight change.* The distribution of fitnesses for different mutation types shows the same relationship as in Figure 4.5 regardless of connectedness (left to right) or the size of weight change (bottom three rows). Each column shows all mutations from initial networks which are set up to have different numbers of connections. A fully recurrent 15 node network with no self-interactions would have 15 x 14 = 210 connections. Here the links are pruned out during initial training to get 52, 105 and 157 connections from left to right. The effect of varying the size of the weight mutation was also investigated. While there is a statistically significant effect here the change in mean fitness is small. For example, in the 75% connected column, between weight changes of 0.1 and 0.5 the reduction in average fitness is only 0.06+/-0.03 (95% confidence range). Although a larger change in weights does give a distribution with more variance, as would be expected.

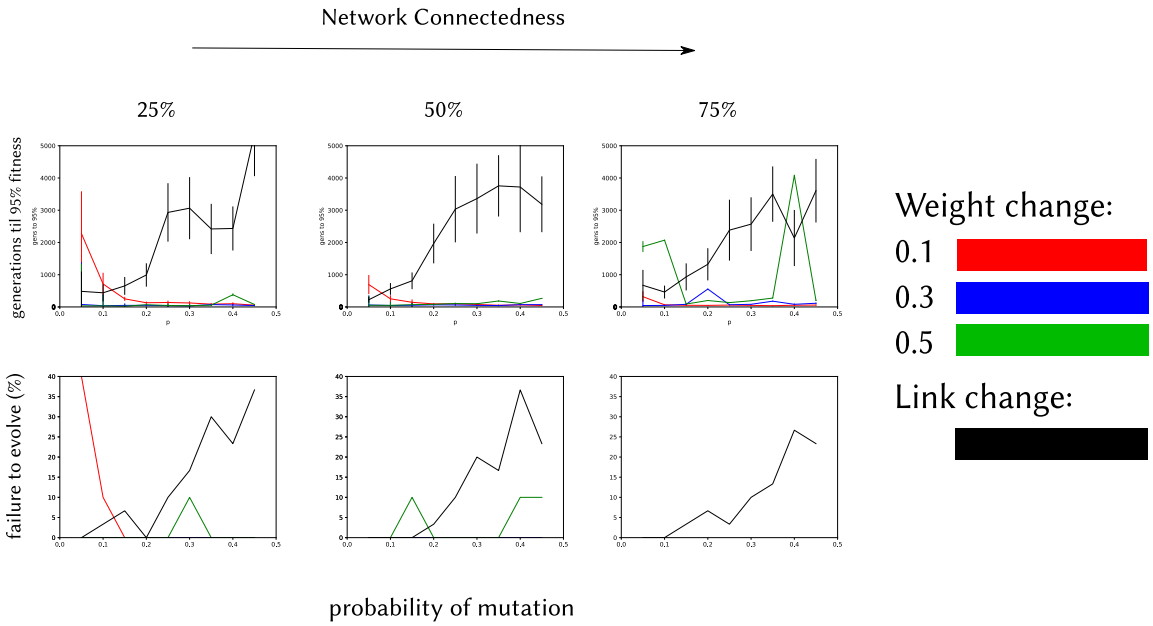


Figure 4.7: *Rate of evolution much slower using link mutations* The top row of figures here show the number of generations it takes to evolve a network to 95% fitness under various conditions. The variable on the x-axis is the probability of mutation, meaning at each generation each weight or link is mutated with that probability. All simulations ran for 10,000 generations, so if a network still hadn't reached 95% fitness by then they are classed as a failure and removed from the data set. The bottom row shows the percentage of evolutionary runs that failed in such a way for each condition. The variability in the time taken for evolution via link-mutations is huge. The error bars show the 95% confidence range of the means (1.96 x standard error). However, even with that variability it is clear that link-mutations led to a much slower evolutionary rate. This result is the same regardless of connectivity or the size of the weight change (red,blue and green lines). As well as being slower to evolve the link-mutations are more likely to lead to failure altogether, not being able to evolve past a local minima.

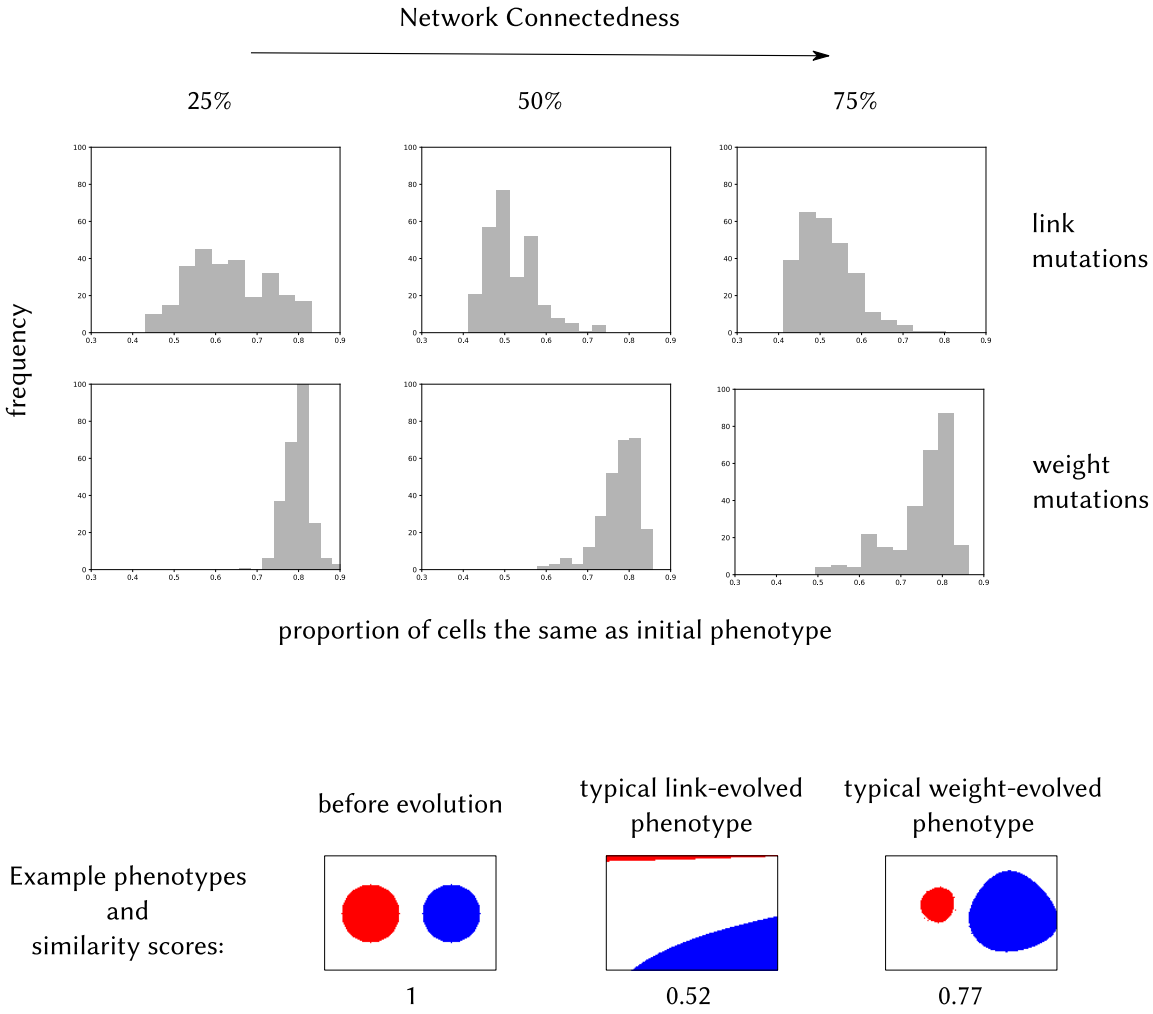


Figure 4.8: *Link mutations cause qualitatively different phenotypes* The evolutionary algorithms are always run until a fitness of 95% is reached, which means the sizes of the blue and red areas are near optimal. However, the positions and shapes of the fields is not accounted for in the fitness function. Evolving the networks with weight mutations automatically conserves the shape and position of the initial phenotype, while just changing the size. Link-mutation evolution leads to drastically different phenotypes even though the size is optimised. This aspect is shown here by counting the number of pixels which are the same colour as the original phenotype, out of a maximum of 15,000. The phenotypes after link-mutation evolutions are much more different to the initial circular fields than those of weight-mutations. The connectedness of the networks has no effect on this result. Each figure shows a distribution of 270 evolutionary runs.

previous results because we need to measure the effect over many mutations rather than one. Due to the time restraints of computation a ceiling of 10,000 generations was put on the evolutionary simulations. If a network took this long and still had not reached peak fitness it is likely because it is stuck in a local minima and would in fact never reach 95%. Therefore these 'failures' are taken out of the data set and plotted separately, giving a second measure of the success of evolution. So of the successful networks, how does mutation type affect the speed of evolution? The graphs clearly show that link mutations cause a much slower evolution, regardless of probability of mutation (x-axis) and connectivity (panels left to right). In addition to this the proportion of failures is also higher for link mutations (bottom row of graphs). The graphs also show that for a small weight change (red lines), the evolution can also be very slow for small mutation probability. This makes sense because the network will change only very slightly each generation. But once the probability of mutation is high enough the size of weight change has little impact on speed of evolution. There is a balance to be struck between making large enough changes to the genotype and searching only the most local space in the landscape.

Thus far we have only judged the results of mutations in relation to a fitness function based on sizes of cortical fields, with no regard for other properties such as position or shape of the fields. It is my assumption that it is beneficial to be able to adapt to a single quality without radically altering other qualities of the phenotype. For example, if altering the size of fields it would be a good property of the evolutionary system to do that with minimal other changes to the phenotype. By looking at the phenotypes produced from link mutations it was obvious that this was not the case. While equally reaching 95% fitness, weight mutations would keep the shape and position of fields the same, whereas link mutations would not.

I measured this effect with the simple metric plotted in Figure 4.8. After all the networks have successfully evolved I measured how different the cells were to the original phenotype, simply by counting what proportion of the 15,000 remained the same. The histograms show that link mutations cause evolved phenotypes that are much less similar (mean of 0.52 the same) than weight mutations (mean of 0.77 the same). The example phenotypes give an obvious sense of the qualitative difference between these cases.

These results clearly demonstrate that different kinds of mutation cause different distributions of fitness, and not necessarily a normal distribution. This has a noticeable effect on the ability of the system to evolve. The Modern Synthesis of evolutionary theory assumes that random mutations will cause a bell-curve of phenotypic fitness, making the system evolvable. However, these results show that not all mutations will lead to such a distribution. Therefore to explain the evolvability of a system it is necessary to know how developmental and mutational mechanisms lead to a useful pool of phenotypes.

4.5 Discussion

By considering the chain of cause and effect which links mutations in DNA to a change in phenotype we can see how developmental dynamics play a role in evolvability. The results of this investigation and the related search of the literature has led me to think that random mutations alone would not lead to a pool of phenotypes which allow a species to evolve. Rather, it is the nature of development and biological configuration which shapes the distribution of phenotypes in a way which

improves evolvability.

The results presented here demonstrate that one type of mutation causes more deleterious effects than another. The distribution of phenotypes for that mutation is not a neat bell-curve and the evolvability is reduced. The problematic mutation involves removing or adding links rather than changing the strength of those links. It has been shown how this mutation is perfectly feasible through either chromosomal mutations or point mutations to the coding-region. Therefore evolvability will increase in species whose development makes these deleterious mutations less probable.

The developmental and genetic systems effects the rate of different mutation in three ways. Firstly, the nature of DNA itself makes deleterious mutations less likely. Because DNA is made up of large regulatory regions and small coding regions there is simply less chance of point mutations in the smaller sequences. Also, the majority of bases in the regulatory regions are not part of binding sites, they are structural, providing spaces between the sites. Secondly, the method of development, using Gene Regulatory Networks to perform thresholding operations, makes the system more robust to a wide range of mutations. Consider that a severe mutation such as adding a whole new binding site amounts to only a small change at the level of the network. I think this is the reason why evolution proceeds mostly by altering the gene networks rather than the proteins themselves (Carroll, 2008). Finally, the biology has many ways of regulating different mutation rates, for example by repairing DNA that has been mutated (Baer et al., 2007). It is known that the mutation rate in mammalian genomes varies enormously, and that this variation can occur at the level of individual sites or whole chromosomes (Hodgkinson and Eyre-Walker, 2011). Many of the mechanisms for such variation are still unknown. Whatever they are it

is safe to say that such regulation holds great importance for evolvability.

In considering the evolvability of the system an important metric is the proportion of phenotypes which offer an improvement. One could argue that the only reason weight mutations led to improved evolvability is that they were more likely to give beneficial phenotypes. Meaning only the proportion of improved phenotypes matters, not the distribution. It is conceivable that if the distribution was other than normal, say uniform, the evolvability could even be increased as long as the proportion of beneficial mutations is high enough. So aside from the fact that normal distributions are more common in nature, as there are more ways to make small changes, are they necessary? This question points to a limitation in our evolutionary model. In our algorithm, each generation is discarded if their fitness does not offer an improvement. However in nature organisms with less fitness than their parents often still survive to reproductive age. At which point the species gets another roll of the evolutionary dice. So to properly investigate this it would be good to use a population model where most of the population can reproduce. Then the effect of distribution on phenotypes (while proportion of beneficial mutations is held fixed) on evolvability can be observed.

The most questionable aspect of the approach given here is whether the neural net accurately portrays the alterations at the gene level. While I have shown that changes in binding strength and number of sites relate to a weight change in the neural network, this largely ignores changes in the logic of the gene. A single mutation could effect the logic in a way which would take a complex combination of weight changes to represent. In the next chapter a new model is developed which better represents the logical nature of these connections rather than using weights, albeit at a high computational cost.

Chapter 5

A novel logic-based artificial gene network

5.1 Introduction

In the previous two chapters a neural network was used to represent the interactions between genes for creating spatial expression patterns. This was done for several reasons, the computational complexity of each node is similar to that of a gene, the ability to train a neural network to reproduce a given pattern is well understood and it was possible to mimic knockout experiments by setting individual nodes to zero. However, if we are to understand how mutations to the genotype affect the phenotype we need a model which is faithful to the biology at the level of those mutations. In the case of a neural network each node's value depends on the others in an additive way. A mutation could effect the strength of a connection and make one node contribute more or less to the total. In real gene networks mutations can

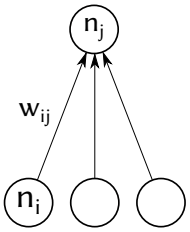
have an additional effect which is difficult to implement with a neural network. They can also change the logic of the interaction non-linearly. For example, substituting a single nucleotide base in a gene could change its logical operation from an AND to a NOT. If we'd like to investigate the effects of an evolutionary algorithm on our model then it should also be capable of making these kinds of mutation.

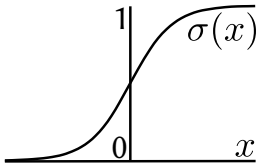
In this chapter I will investigate the characteristics of gene networks for cortical patterning, using a novel model which makes use of the discrete logical nature of real genes. The aim is to find a minimal network structure and parameters which allow the evolution of a cortical field. The fundamental idea behind this model of pattern formation remains the same as in the previous model. Each cell in a sheet is using the same computational network, encoded in its DNA, to perform a thresholding operation. Each cell thus gives a different expression level output depending on its position according to several gradient inputs. Viewing all cells together this appears as a boundaried shape, for example the primary visual field in cortex.

The crucial difference in this new model is that the combination of input nodes will be through a Boolean logic rather than a summation. This allows us to make a new kind of mutation in the evolutionary algorithm but takes away the ability to train the network by back propagation. One difficulty with developing such a model is combining the discrete nature of the logic with the continuous nature of the gene expression levels (transcription factor concentration). This is achieved by using the Hill function to find the probability of a single binding site being bound, then combining these probabilites using the Sum of Products of a truth table. A comparison of the two network models can be found in Figure 5.1.

In this introduction I will firstly summarise the biological mechanics of how mu-

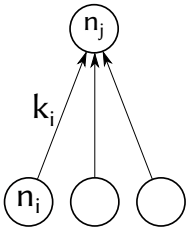
Neural Network

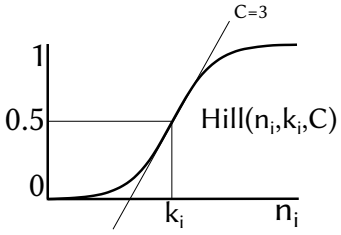


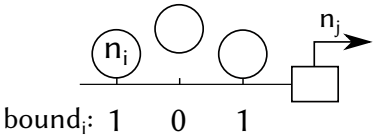


$$n_j = \sigma(\sum_j w_{ij} n_i)$$

'Logic' Network







$bound_i$	TT_{output}
0 0 0	0
0 0 1	0
0 1 0	1
0 1 1	0
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	1

$$n_j = \sum_{row}^{2^N} (TT_{row} \prod_i^N f(n_i)) \quad \text{(Sum of Products)}$$

$$f(n_i) = \begin{cases} Hill([n_i], k_i, C_i) & bound_i = 1 \\ 1 - Hill([n_i], k_i, C_i) & bound_i = 0 \end{cases}$$

Figure 5.1: The 'Logic' Network computes the probability of gene transcription in a very similar way to how real genes do. The gene transcription is on or off (1 or 0) depending on the combination of binding sites which are occupied. The result of all possible combinations is shown as a truth table. Each element in the truth table happens with a probability calculated using the Hill function. The total probability is the sum of products of the truth table. Each column of the truth table represents a binding site with differing binding strength, k_i, and cooperativity factor C_i.

tations effect gene networks. This will inform the design of the new model. Then finally the possible research aims of using such a model will be discussed.

5.2 Biological background

The biology of how mutations affect gene regulatory networks for spatial patterning has been covered in a lot of detail in the previous chapter. The most important aspects to briefly recap here are about how a gene computes Boolean logic, and what determines the probability of transcription.

The first step to understanding how DNA mutations relate to changes in the gene network is to know that a single gene, the DNA sequence responsible for an individual protein, has two main parts. There is a section which codes for the protein, giving the order of amino acids necessary to build the protein, and there is a non-coding section, also called the cis-regulatory element. The cis-regulatory element is so called because it regulates the transcription of the coding element and sits on the same strand of DNA just next to it. In the coding section mutations to the DNA will effect the protein that the gene builds, in the non-coding section mutations will alter the regulation of gene-transcription, meaning the circumstances under which the protein will or won't be built are altered.

The effects on regulation that can occur from a single mutation are extremely varied and complex. The sequence of DNA base pairs in the non-coding region includes sections which proteins can bind to with other sections inbetween that simply space out these binding sites. The proteins that bind to the binding sites are known as transcription factors, this is because the act of their binding to the DNA effects

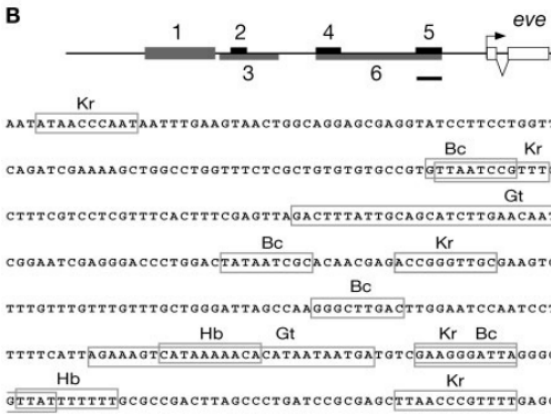


Figure 5.2: Binding sites for transcription factors Kruppel, Giant and Bicoid overlapping in the 2nd cis-regulatory module of the eve gene. The bases in between the binding sites determine their spacing, which effects their mutual interactions. Clearly simply inserting, deleting or substituting individual bases can have an enormous range of effects on the whole. Figure taken from (Wray et al., 2003), their Figure 3B.

the ability of the transcriptional machinery in the coding part of the gene to begin transcription. So a mutation to the DNA (say an insertion, deletion or substitution of base-pairs) will effect these binding sites in various ways and thus the regulation of transcription. The binding sites are not independent, the act of one transcription factor being bound may effect others, either through weak hydrogen bonds or allosteric binding to other transcription factors. The overlapping of binding sites also means that it can be quite simple for one transcription factor to repress the binding of another. See figure 5.2 taken from Wray et al. (2003) for an example.

The probability of binding as a function of the concentration of TF at each individual binding site can be modelled as a Hill function. The relevant parameters which effect the function are the strength of attraction of a TF to the site, and the cooperativity between that site and adjacent site. Modelling a collection of binding

sites requires taking into account that their positioning (ordering and spacing) effects the boolean logic of the regulatory element, simply moving one binding site slightly further away from another could turn a NOT function into an AND function. Taking these two elements together we can model a single genes transcription by a set of Hill functions combined with a certain Boolean logic. Mutations will correspond to changes in either of the two Hill parameters or to the logic of the whole.

The present work aims not to account for every possible mutation and its corresponding change in the model, but rather get a sense of the range of biologically feasible alterations with a single mutation. It has been shown that by simply changing the spacing between sites, thus not relying on the more complex allosteric chemistry, the full range of boolean logic operations can be implemented by a single bacterial gene (Buchler et al., 2003). Therefore it is an assumption of this model that a single mutation can make any possible change to the boolean logic, although feasible it is not necessary to argue that this is likely.

5.3 The Sum-of-Hill-Products Model of Gene Transcription

Gene transcription occurs based on the combination of binding sites which are bound. Say there are five binding sites, and labelling each site with a 1 represents being bound, and 0 means not bound. Then each combination will give an output of transcription or not, deterministically. For example, the combination 10101 may cause transcription, while 11000 does not. Taking these combinations together produces a truth table representing the logic of the cis-regulatory module. We can also describe

an exact function for the likelihood of an individual site binding dependent on ligand concentration, this is the Hill function. If we assume the probability of binding is independent (which it is really not), then the likelihood of a particular combination, say 10101, is the product of five Hill functions (or 1 minus Hill in the case of zeros). Now there is not only one combination that allows transcription, so to find the total probability of transcription we have to add every possible combination which causes transcription together. In the truth table this means summing the rows with an output of 1. Therefore, the final probability of transcription is the Sum-of-Products of the truth table, where each 1 or 0 is replaced by the likelihood of binding or not (the Hill function). Each column of the truth table has different parameters for the Hill function depending on binding strength (dissociation constant) and effect of other local sites (cooperativity).

We know that real mutations to the cis-regulatory module will change either the binding strength of an individual site, or the logic of the truth table. We can use this in an evolutionary algorithm to train the network to produce a desired output. Starting with the concentration of 4 genes already constrained, by forming 4 orthogonal/opposing gradients, we can add another 4 genes and pick one of them to represent the output. After each change to the network via mutation the error is measured (the absolute difference between target and output summed over many positions in the space). If the error is the best found so far the network parameters are saved and mutations continue from that new optimum, gradually the error decreases and the network ends up producing the target image. This was successfully achieved for creating a single circular region of high expression in the middle of the image, it did so more accurately than a neural network of the same size and structure.

The fundamental idea behind this model of pattern formation remains the same

as in the previous model. Each cell in a sheet is using the same computational network, encoded in its DNA, to perform a thresholding operation. Each cell thus gives a different expression level output depending on its position according to several gradient inputs. Viewing all cells together this appears as a boundaried shape, for example the visual field in cortex.

5.4 Research aims

The primary aim of this project is to develop the model described above, using it to produce a desired expression field via an evolutionary process.

The ability of the 'Logic' network to evolve to a configuration where it produces a single cortical field will be observed under different conditions. The main variable under consideration is how many input nodes with gradients the network has, 2, 3, 4 or 5. In the case of two inputs the nodes will have orthogonal linear gradients, theoretically enough to give a coordinate system. With three inputs the system is similar to that of FGF, WNT and BMP secretion in the early cortical plate, diffusing from three equadistant points. With four inputs the system is similar to that of COUP-TF1, EMX2, PAX6 and SP8 with the original two gradients plus their opposites. We will also test 5 inputs for completeness.

We will also investigate the success of evolutionary trials under various mutation rates, or probability of mutation. There are two types of mutation in the evolutionary algorithm, one effecting the logic of the network connections by flipping individual bits in the truth table with probability p_{TT}. The other effects the binding strength of each site by altering that strength to another possible strength in the same with

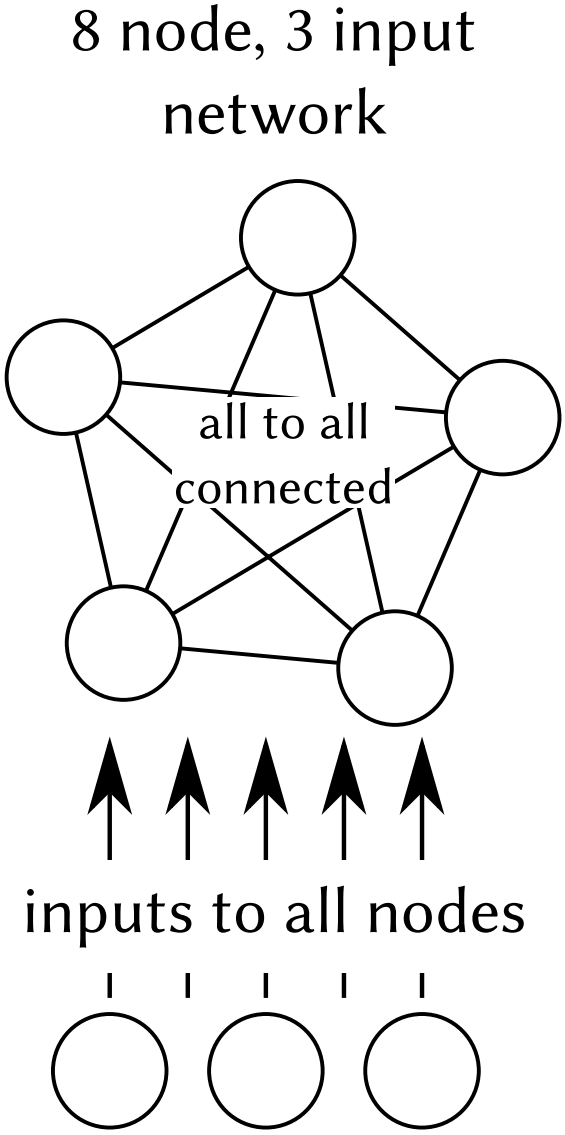


Figure 5.3: Each node value represents the concentration level of a gene product. The three input nodes 'bind' to the regulatory element of all of the 5 recurrent nodes. One of the recurrent nodes is the output concentration value, which is what determines the error of the network depending on the values of the inputs. As the network evolves the connection's strength and logic is altered and the error decreases. As the values of the 3 input nodes are spatially monotonic, creating a coordinate reference, the output values over the whole domain represent a spatial pattern.

probability p_B. These mutations are carefully chosen to reflect real mutations that occur in the cis regulatory module and the ability to use such mutations is in fact the motivation of using this Logic system at all. Using the previous model of the neural network, mutations either effected the existence of connections between nodes or their weights. It was found that mutating the weights rather than completely adding or removing links made the system much more evolvable, however it was not possible to draw a direct analogy from these mutations to those in the biology.

Under all of these conditions there will be 10 runs of 2000 generations each. It is possible that the evolution gets stuck in a local minima and the error stops decreasing, if it does this for 100 generations the trial is stopped and counted as unsuccessful. Therefore the proportion of successful trials, and the number of generations taken for success will be the observations for our analysis.

5.5 Methods

The network is setup to have an input layer and a recurrent layer. If we hold the total number of nodes to be a constant 8, then as the number of inputs increases the number in the recurrent layer will have to go down. See the Figure 5.3 as an example. To be rigorous in the investigation therefore we will test networks with 8 nodes in total, and also networks with 4 recurrent nodes constant, eg 5 inputs + 3 recurrent (8 total) and 5 inputs + 4 recurrent (9 total).

Each network is initialised with a random truth table and all binding strengths, k, set to 10nM. Cooperativity, C is constant for all binding sites as 3.

For each generation a process of mutation, network settling, error measurement and selection is repeated. Mutations to the truth table involve taking each output value in turn and with probability p_{TT} subtracting it from 1 (flipping the bit).

The binding strength mutation involves each binding site changing with probability p_B, the new strength is chosen randomly from a sample of k values from empirical data with Drosophilia Jung et al. (2018), shown in Figure 5.4.

After mutating the network to generate a new candidate we must iterate through the settling algorithm to find the output for any set of inputs. The aim is to reproduce a 10 pixel radius circle in the centre of a 50x50 pixel field. For each pixel the values of the inputs differs, the map of coordinate to input value depends on the number of inputs as shown in Figure 5.5. So for each coordinate (x,y) the inputs are set, the signal propagated through the network according to the equations in Figure 5.1, then the average output value over the final 10 settling steps is taken. It is possible to settle the network until convergence then take the final value, but occasionally the network lands in a limit cycle rather than converges, so averaging avoids making exceptions for unusual cases.

Now the output value for (x,y) is found it can be compared to the target output (1 within the circle radius, 0 outside). This error value is totalled up for all 50x50 pixels, if the candidate network has equalled or bettered the previous network it is accepted as the new standard to which subsequent mutations are performed.

Not all networks successfully evolve to create the desired pattern, some get stuck in local minima. If the networks get stuck for 100 generations or if the error is not low enough after 2000 generations the trial is counted as unsuccessful. By optimising parameters the maximum success rate I have found is 80%.

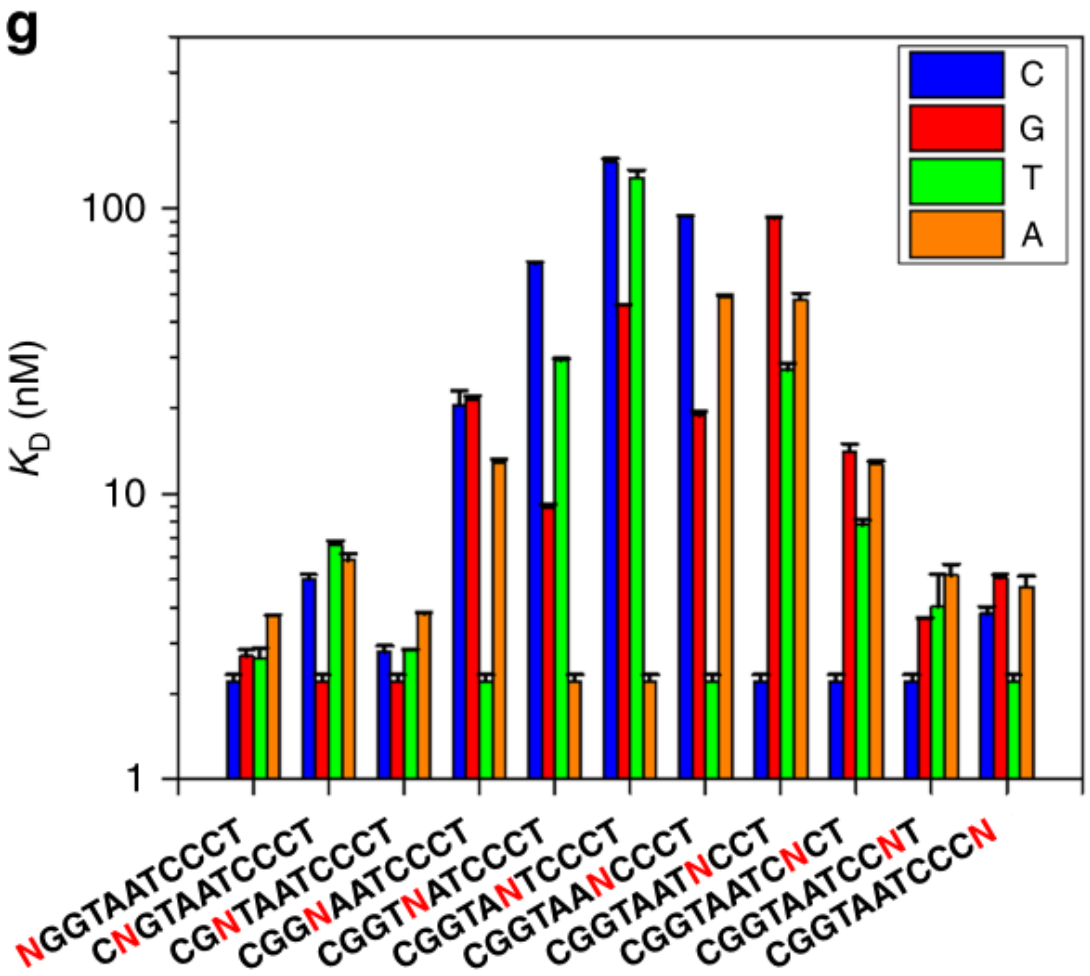


Figure 5.4: From Jung(2018), the dissociation constant, k is empirically given for every possible mutation to a single binding site. By sampling from this distribution I mimic the change to binding strength thanks to a single base substitution.

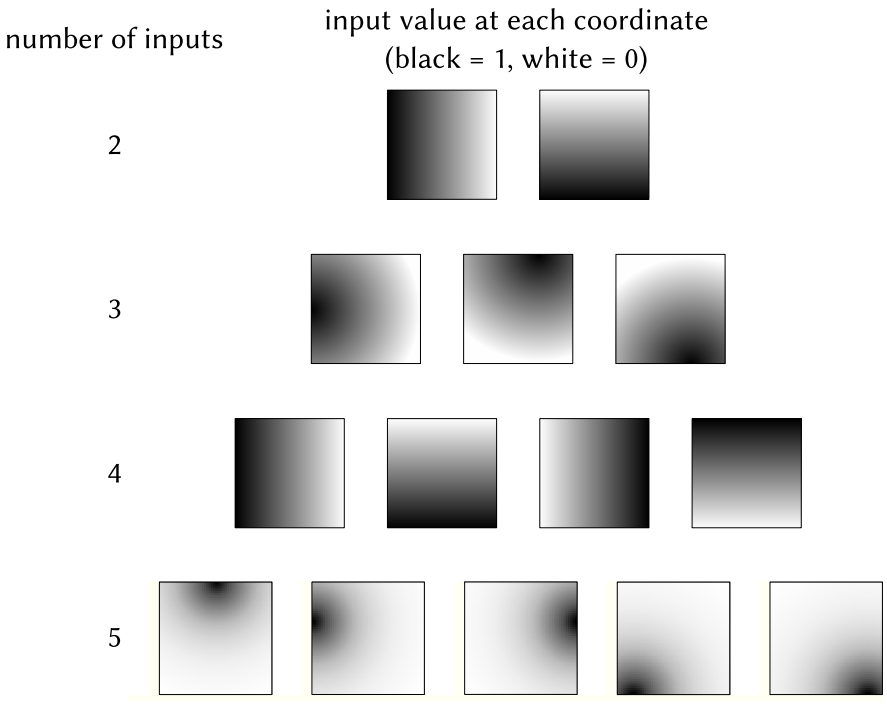


Figure 5.5: The darkness of each pixel represents the value of that input node at that coordinate. This shows the input mappings for 2,3,4 or 5 inputs. The mappings for 3 or 4 nodes is similar to that found in the cortex, for 2 or 5 we have extended the principle of diffusing gradients regularly in all directions.

5.6 Results

This model portrays the development of spatial patterns in a more biologically plausible way, and requires a similar number of nodes to a neural network. Each node has more potential logical operations than in the neural net model, so a single field can be created with just 8 nodes. How those 8 nodes are set up in terms of the proportion that are input nodes is important to the success however. The main result is that the success rate (i.e. fraction not getting stuck in a local minima) depends on the number of linear gradient inputs. 4 linear gradients x, y, 1-x and 1-y, gives the best chance of success. This may have some bearing on the reason for the cortex using four genes as a coordinate reference. When the total number of nodes is 8 the average success rate for 3,4 or 5 nodes is 2.9, 4.4 and 3.8 out of 10 respectively, see Figure 5.6.

In the simulations two parameters were varied, as well as the number of inputs. The mutation step in the evolutionary algorithm can either change the truth table, or the binding strength of individual binding sites. Each output bit of the truth table can flip with probability p_{TT}, each binding site changes its strength to another from a realistic sample distribution with p_B. Probabilities in the range 0.005 to 0.06 were tested with 10 trials at each 0.005 interval, it was found that evolution was markedly less successful for $p_{TT} > 0.035$. There was much more variance in the success rate as p_{TT} varied, with p_B having less of an effect. See Figure 5.7

The same trials were run for networks with 4 recurrent nodes, so 7,8 or 9 nodes total for 3,4 or 5 inputs. The results were almost identical to when constraining to 8 nodes total. However it was found that 5 inputs (9 nodes) performed slightly better than 4 (8 nodes), with an average of 4.6/10 compared to 4.4. This small improvement

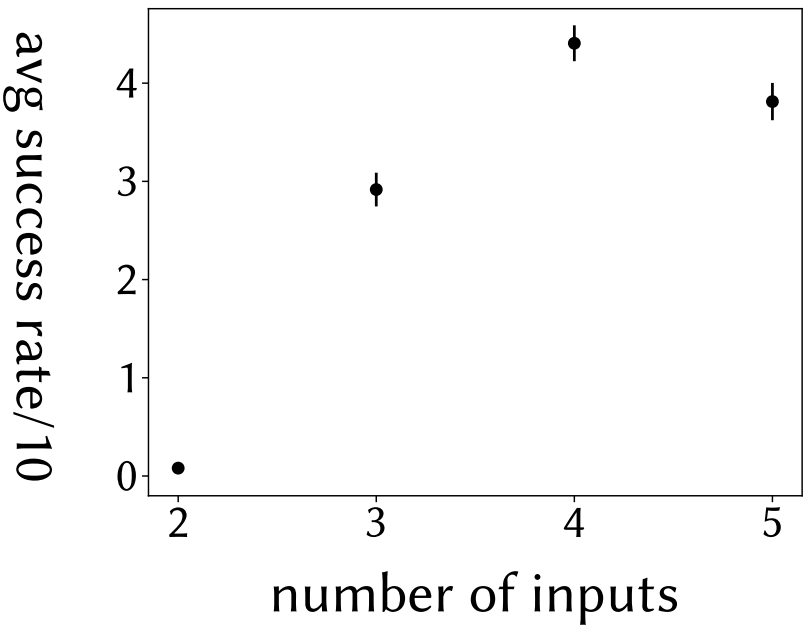


Figure 5.6: Averaging the success rate over all trials where $p_{TT} \leq 0.04$ and $0.05 \leq p_B \leq 0.06$, which is the domain where there is a reasonable rate of success for 3,4 or 5 inputs with 8 nodes total. We find that for 4 inputs the evolutionary algorithm is more likely to find a solution than for 3 or 5 inputs. Networks with only 2 inputs barely ever found a configuration which produced the single circular field, however the success rate is non-zero so it is possible. The error bars are standard error, the sample size was 96 for each bar.

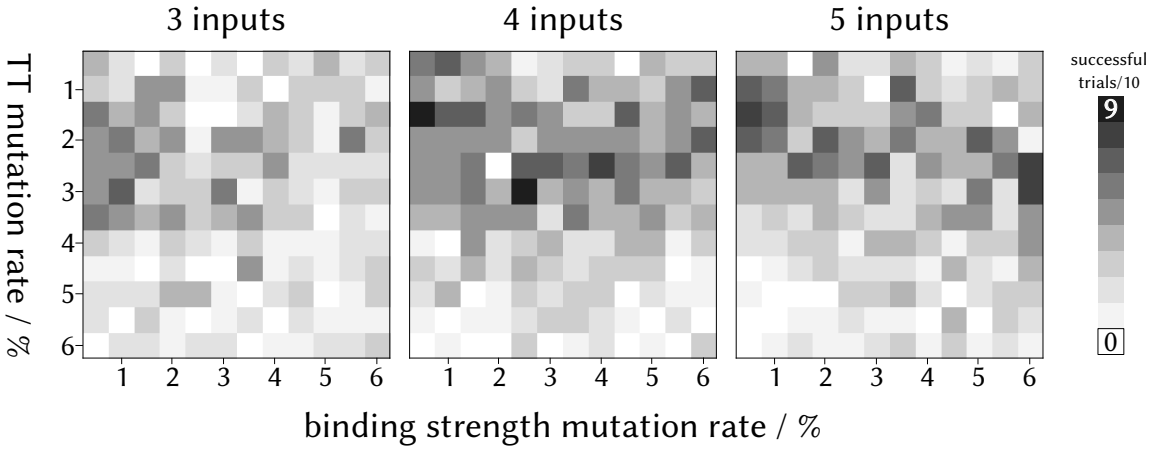


Figure 5.7: For networks with 8 nodes total. The darkness of each pixel in these figures represents the likelihood of successfully evolving a solution dependent on variables p_{TT}, p_B and number of inputs. There is little variance across values of p_B but success is much more sensitive to the value of p_{TT}, dropping off rapidly around p_{TT}=0.04.

required a disproportionate amount of extra computational power as the single extra node causes the truth table to double in rows. See Figures 5.8 and 5.9.

5.7 Conclusion

In the previous chapter on neural networks we tried to infer general network characteristics from the statistics of successful networks. However, the method was unfortunately not enough to infer exact gene interactions responsible for an empirical phenotype. Perhaps this new model, as it is more biologically realistic, could give a more exact picture of gene interactions if we used it to reverse engineer known phenotype patterns including knockout experiments. Here we face a new problem, the computational intensity of this new model. To use this model we had to make

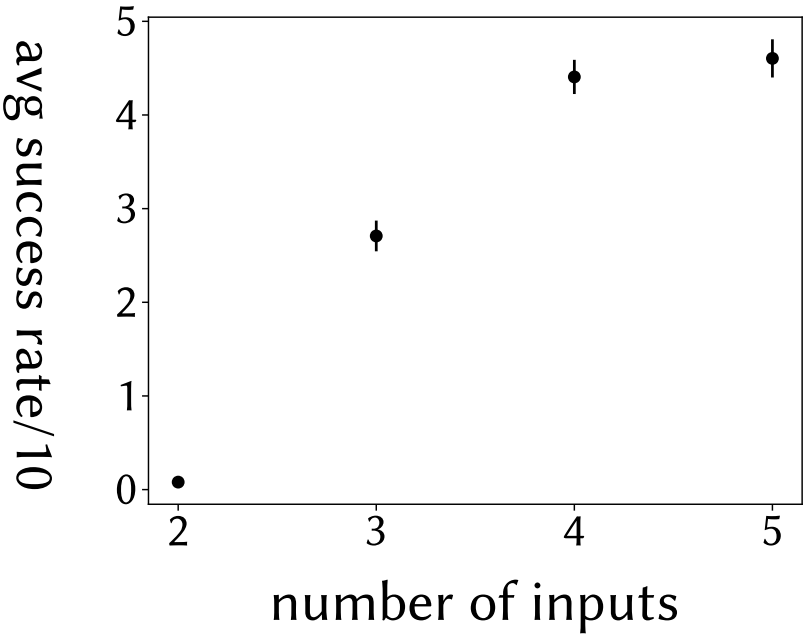


Figure 5.8: Averaging the success rate over all trials where $p_{TT} \leq 0.04$ and $0.05 \leq p_B \leq 0.06$, which is the domain where there is a reasonable rate of success for 3,4 or 5 inputs with 4 recurrent nodes. So now the total nodes are 7, 8 or 9 respectively. In this case there is a marginal improvement for 5 inputs, but not proportionate to the extra computation required. The error bars are standard error, the sample size was 96 for each bar.

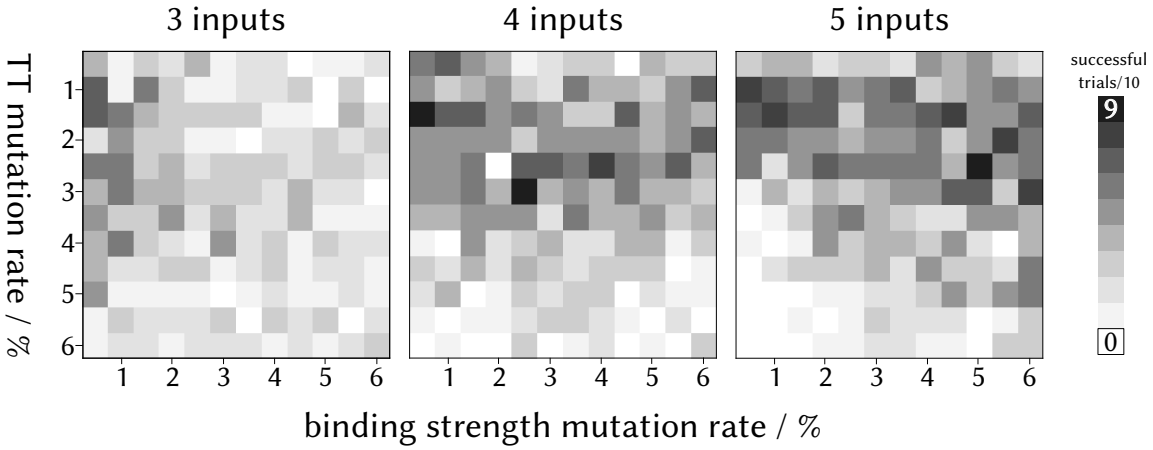


Figure 5.9: For networks with 4 recurrent nodes but varying total nodes. The darkness of each pixel in these figures represents the likelihood of successfully evolving a solution dependent on variables p_{TT}, p_B and number of inputs. There is little variance across values of p_B but success is much more sensitive to the value of p_{TT}, dropping off rapidly around p_{TT}=0.04.

several simplifications, the region of pixels was smaller, the number of nodes only 8, the phenotype only a single field. Is there a way we could produce a complex phenotype such as mouse neocortex without an overly lengthy computational time. How long would the compute-time be? After all leaving the computation running for a month would still be feasible if we could be certain that it was computing something meaningful.

The main issue is requiring more nodes, firstly because it simply increases the computation needed to process the truth table, but also because it increases the size of the space of possible networks, throwing extra hay onto the needle to be found.

One method of reducing the compute time is by using sparse networks, as the rows of the truth table grows as 2^n where n is the number of inputs. This is a reasonable simplification as real gene networks are also sparse. But if the aim is

to reverse engineer the real gene interactions how can you be sure not to prune an interaction which exists in the real biology?

I hope this chapter has presented what is at least a promising new model of gene regulation which can be taken further in follow up work. We have taken the core principles which govern real gene transcription and shown that the system is evolvable using realistic mutations. While computationally intensive the number of nodes required is similar to those of real gene networks for spatial patterning, i.e. less than 10.

Chapter 6

Discussion

The question I have been addressing in this thesis is how the structure of the genotype and its mapping to phenotype affect evolvability. In particular, the system I have focused on is the gene networks responsible for cortical arealisation. What I have learnt throughout the course of these projects is how crucial development is to the evolvability of a system. This is what structures the genotypic space.

The chapters presented here have been about understanding how elements such as DNA, mutations, genes and network dynamics fit together to create a pool of phenotypes which natural selection can act on. These parts can be abstracted in the following way. There is a genotype space which consists of every possible genome, and it is possible to move from one to another by a single mutation. The process of development then maps each single genome to a phenotype. Each phenotype has some probability of success depending on the environment, determining the replication of its genome into future generations. The ability for this process to work properly, that is to be evolvable, depends on two things. Firstly, the graph of genomes must

be connected in such a way that it is possible to get from one point to another via many different paths. This is important because there is no telling which genomes will be damaging to the success of the phenotype depending on the environment, so there must be lots of options. Secondly, there should be good correspondence between genotype and phenotype, so that local genomes map onto phenotypes of similar fitness. Gene networks accomplish both features, making systems highly evolvable.

The ability to traverse the genotype space in many ways is linked to the mathematical phenomena of 'percolation' (Gavrilets and Gravner, 1997), or in evolutionary science, 'neutral networks'. There are many ways that the DNA system is set up to create neutral networks, so that mutations have zero effect on the fitness of a genome. Kimura (1968) found that neutral substitutions must be the most common form of mutations, otherwise mammals would not be able to tolerate the high rate of mutation they experience. The fact that there is so much redundancy in the DNA encoding is one simple way that neutral networks are created, gene networks have a similar but more complex role.

Gene networks involve many layers of self-organisation, each one creates more distance between mutations and their effects on development. Each layer also has its own redundancy which gives opportunities for neutral networks to exist. In Chapter 4 I went into detail about all the possible effects of mutations on gene networks. Aspects which help structure the genotype space well include the following. It is much more likely for a mutation to occur in the non-coding region of DNA rather than the coding region, so that the regulation of a protein is effected rather than the protein itself. Within the non-coding region, it is much more likely for a mutation to affect the space between binding sites rather than the sites themselves. Changing the spacing affects regulation in a smooth way by altering the cooperative binding probabilities.

If there is a mutation within a binding site it will also have an incremental effect, by changing the binding strength of a transcription factor rather than altering the type of site it is. The non-coding region of the gene is also modular, so if the logic of one module is drastically changed there are still other modules that perform the same computations as before. Finally, each module has a lot of binding sites, several for each input transcription factor, so if a single site is changed it will likely not have a severe impact on the logic.

Self-organising systems have the characteristic of simplicity emerging from complexity. They make use of energy in the environment to create order from many interacting parts. A striking example in this case is the self assembly of transcription factors and regulatory proteins at DNA binding sites, which work together in complex ways to decide one of two states, transcription being on or off. Then at a higher level, many genes affect each other's activity in a recurrent network which settles into an attractor state. The phenomenon we see arising again and again in these systems is an almost infinite complexity leading to relatively few emergent possibilities. In terms of the abstraction of the genotypic space this is seen as a highly connected, many dimensional network of genomes which maps onto a smaller number of phenotypes.

The first project presented here took a very simplified model of cortical arealisation and showed how self-organisation led to attractors which structured the fitness landscape in a way which sped up evolution. From there I made models which increasingly captured the real systems of self-organisation in the developmental process. Eventually we ended up with a model where we could make mutations to the genotype in a very similar way to real genomes and study the local structure of the space along with the systems evolvability. I'll now review each part of the thesis and

how it contributed to this overall story.

6.1 Part 1 – Boolean Networks

The Boolean Network is a simple abstraction of gene networks. Each gene can have only one of two states, on or off. Its state at each time step is determined by the states of all the other genes at the step before. We can consider the phenotype of the system to be the pattern of gene expression once the network settles down, development to be the network dynamics, and the genome to be the code which determines one time step from the one before. The genome is a string of 1s and 0s which form a truth table, or state transition table. If there are 5 genes there are $2^5 = 32$ possible states that can exist, but $2^{(5*2^5)} \approx 10^{48}$ possible state transition tables. Thus even with 5 genes the genotype space is unimaginably large. Although this means there are myriad ways the network dynamics can settle down, the total number of eventual attractors is tiny. The genotype space is also extremely compact yet many-dimensional, meaning it is easy to get from any one of the 10^{48} genomes to another one. Think about four corners of a square, the largest distance between two corners is just two edges, and every corner is linked to 2 others. The genotype space here is a square in 160 $(5 * 2^5)$ dimensions, so all 10^{48} states are a maximum of 160 edges away from any other, and linked to 160 others.

While boolean networks are an over-simplified model of real genes, they capture the most fundamental aspects of the genotype space relating to evolution. The research published shows that this leads to the evolutionary process being much faster than a random search. Real gene networks also structure the genotype space

to be compact and highly dimensional. The rest of the projects in this thesis aim to explore this by improving the biological similarity of the models. The problematic features of boolean networks that need improving are the binary nature of genes and the lack of scalability to more complex spatial patterns.

In this first project the spatial pattern is simply 2 cells, each of which is either on or off. These represent the anterior and posterior of the cortex. Each cell has its own target for development, for example for 5 gene states to move from '10000' to '10101'. A 'gradient' in gene expression would simply be represented by a '1' in the anterior cell and a '0' in the posterior. This poses some obvious problems, firstly if you wanted to increase to 3 cells how would you represent a gradient, neither '1,1,0' or '1,0.5,0' is a good option. Secondly, as each gene only has two states you need a large number of genes to give many different developmental pathways to different cells, with n genes you could only distinguish 2n cells. With a continuous system you only need 1 gene in a smooth gradient so that thousands of cells are slightly different to their neighbours. Which relates to the third issue, the locality of the cells is not used at all by the boolean model. If there was a 2D grid of cells each would have to be initialised with no relevance to its neighbours.

Therefore the aim of the next project was to make a model which can evolve to create complex 2D spatial patterns, while still capturing those fundamental aspects of the genotype space.

6.2 Part 2 – Reverse engineering Gene Nets for cortical arealisation

A neural network model was chosen as a basis for creating evolvable 2D spatial patterns. Although such models were invented to mimic neurons rather than genes, the computational ability of each node in the network is similar in both cases. The inspiration for using a neural network to model gene networks is really thanks to the non-linearisation of the sigmoid function. In biological systems the very similar looking Hill function is often used. In a neural network the input of one node to another saturates due to the sigmoid, in a gene network the same occurs thanks to the probability of binding approaching one.

Whilst we developed the model to explore evolvability, neural networks have the added benefit that they can be trained via supervised learning. Thus there was an opportunity to train the networks to reproduce known cortical patterns and analyse the commonalities amongst successful networks. Furthermore, it was possible to associate specific nodes with known genes involved in cortical arealisation in mice. By setting a particular node to zero and then training the network to the cortical pattern for when a gene is knocked out, the resultant trained networks hold information about that gene.

The results provided confidence about the ability of this model to capture the essence of the thresholding operations performed by real gene networks. By training many networks to reproduce the cortical pattern in Coup-tf1 knockout mice, we successfully predicted the direction of Coup-tf1's expression gradient. Also, the shapes within the model expression patterns at different levels of the network accu-

rately reflect real expression patterns, with linear gradients at lower levels and well boundaried shapes closer to the final area markers.

At this stage we had developed a model which can produce complex two dimensional expression profiles, using a computational process analagous to real gene networks. However, there was no certainty that the parameters for a successful network could be attained through an evolutionary process. The next step was to do just that, giving us the ability to use this more accurate model to study evolvability and its relation to the genotype space.

6.3 Part 3 – Evolvability due to different mutation types

In Chapter 4 I discussed how a neural network can be 'mutated' in two different ways. Each weight could be changed by a small amount according to an arbitrary probability. This would be equivalent to the small changes in the regulatory section of a gene. Otherwise, entire links in the network could be added or removed. This much more severe mutation would also be possible for a real gene but is less likely.

The analysis in this part ties together two aspects of evolutionary development and explores them with reference to our neural network model. The first aspect is the distribution of fitnesses local to a particular genome. It is an important feature of the Modern Synthesis of evolutionary theory that this distribution be a normal 'bell curve'. The local genotype space is determined by what mutations are available, each genome is one step away from others via a single mutation. Therefore if a different set

of mutations is available then the local distribution may or may not be appropriate for an evolutionary process. The second aspect is a measure of evolutionary success. Under each mutational regime the number of generations for the phenotype to evolve from one state to another is measured.

We found that with 'weight' mutations the local genotype space was indeed neatly normal, however with 'link' mutations it was not. Link mutations were also much more likely to produce a poor fitness. Evolvability is defined by the availability of adaptive variation, so we would say that a system which prioritises these weight mutations is more evolvable. This is borne out by the fact that the measured number of generations to evolve was much fewer for the weight mutation system.

In the beginning of this discussion I hypothesised that evolvability requires at least two features of the genotype-phenotype space. This study clearly demonstrates the second, that a local genotype space corresponds simply and smoothly to a local phenotype space, with a normal fitness distribution. There is no reason why an artificial system has to have this feature, this study shows it is possible to design mutations which make the correspondence less successful. However, it is empirical truth that natural systems do have this feature, which is why Fisher assumed a normal distribution of fitness in the first place. The genetic system has itself evolved. Systems that confer less evolvability will themselves not be replicated as successfully. My argument here is that gene networks have evolved into existence because they succeed at making the system evolvable. They are successful because the mutations which are available to them correctly structure the genotypic space.

6.4 Part 4 – Modelling mutations in cis-regulatory modules

While the previous study shows the importance of types of mutation for the evolvability of a system, it is lacking in its accuracy of modelling real gene mutations. The issue is that weight and link mutations in the neural network, while accurately reflecting some possible gene mutations, ignore others. For example, a real cis-regulatory module of a gene can be understood by its discrete boolean logic. This logic is a result of the placement of its various binding sites. It is possible that even a slight change in the positions of binding sites would cause a discrete shift in the logic of transciptional regulation. If it was the case that such a mutation would structure the genotype space differently to the weight mutations of the neural network, I'd have to abandon my argument from the previous section. For this reason I developed a new model which can combine the continuous complex pattern formation of the neural networks, but also be governed by a discrete boolean logic.

The model combines the continuous nature of protein concentrations with the discrete logic of transcription regulation. A binary truth table governs the logic, while the probability of any row in that truth table occuring is a continuous function of protein concentration.

Again in this study there are two types of mutation investigated. Both of these are highly realistic with reference to gene regulation. The first involves changing the binding strength of a particular binding site, which is what happens when a single nucleotide base is substituted within a site. The range of binding strengths used comes from real data of a Drosophilia gene when subjected to every possible

nucleotide substitution. The second kind of mutation deals with the discrete logic of regulation, by flipping some of the binary elements of the truth table.

This model requires a lot more computational power than the neural net model, so the number of genes we are able to simulate is much lower. For this reason the target phenotype for this network was only a single cortical field. We investigated the ability to evolve dependent on the relative prevalence of the two types of mutation. Finding that the rate of evolution is much more sensitive to the discrete logic mutations, with binding strength mutations having little impact on evolvability. At higher rates of logic mutations the system becomes less capable of evolving.

As this model better represents the computational ability of each individual gene I reimplemented some earlier investigations into the minimal structure required for a successful network. In particular the number of input gradients is of crucial importance to the success of forming a single field. It is theoretically possible to do so with only two orthogonal inputs, as we might expect for a Cartesian reference frame. However the rate of evolution is much improved by having four inputs, and to a lesser degree for having five. This provides a possible reason for why we see four gradient inputs in the cortex (Emx2, Pax6, Sp8, Coup-tf1), while the positional information theory of Wolpert would only require two.

6.5 To Conclude

The studies presented in this thesis show a process of gradual refining of understanding through one system of development. The developmental system of cortical arealisation is relatively neat from a mathematical point of view, compared to other

developmental systems. The existence of the protomap very early in development allows us to abstract the problem as a pattern of two dimensional gene expression. Yet, as we have discussed in this chapter it is the immense complexity of these systems that improves their evolvability. Through several levels of non-linear emergent phenomena the genotype is structured as a high dimensional space, which means a rich pool of variation is always available. I greatly look forward to seeing how these models at each level of abstraction - DNA, the gene, the network - improve in the future, and how they might be used to reveal the real gene interactions underlying cortical development and evolution.

Bibliography

R. Albert and H. G. Othmer. The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in Drosophila melanogaster, 2003. ISSN 00225193.

N. Antón-Bolaños, A. Espinosa, and G. López-Bendito. Developmental interactions between thalamus and cortex: a true love reciprocal story. *Current Opinion in Neurobiology*, 52:33–41, 2018. ISSN 0959-4388. doi: https://doi.org/10.1016/j.conb.2018.04.018. URL `http://www.sciencedirect.com/science/article/pii/S0959438817302957`.

Y. Arimatsu, M. Miyamoto, I. Nihonmatsu, K. Hirata, Y. Uratani, Y. Hatanaka, and K. Takiguchi-Hayashi. Early regional specification for a molecular neuronal phenotype in the rat neocortex. *Proceedings of the National Academy of Sciences*, 1992. ISSN 0027-8424. doi: 10.1073/pnas.89.19.8879.

M. Armentano, S.-J. Chou, G. Srubek Tomassy, A. Leingärtner, D. O'Leary, and M. Studer. COUP-TFI regulates the balance of cortical patterning between frontal/motor and sensory areas. *Nature Neuroscience*, 10(10):1277–1286, 2007. doi: 10.1038/nn1958.

H. L. Ashe. The interpretation of morphogen gradients. *Development*, 133 (3):385–394, 2006. ISSN 0950-1991. doi: 10.1242/dev.02238. URL http://dev.biologists.org/cgi/doi/10.1242/dev.02238.

G. Ashkenasy and M. R. Ghadiri. Boolean logic functions of a synthetic peptide network. *J. Am. Chem. Soc.*, 126:11140–11141, 2004.

S. Assimacopoulos, T. Kao, N. P. Issa, and E. A. Grove. Fibroblast growth factor 8 organizes the neocortical area map and regulates sensory map topography. *Journal of Neuroscience*, 32(21):7191–7201, 2012. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.0071-12.2012. URL http://www.jneurosci.org/content/32/21/7191.

M. Bachler and A. Neubüser. Expression of members of the Fgf family and their receptors during midfacial development. *Mechanisms of Development*, 2001. ISSN 09254773. doi: 10.1016/S0925-4773(00)00518-9.

C. F. Baer, M. M. Miyamoto, and D. R. Denver. Mutation rate variation in multicellular eukaryotes : causes and consequences. 8(August):619–631, 2007. doi: 10.1038/nrg2158.

P. Bak. *How Nature Works*. 1996. doi: 10.1007/978-1-4757-5426-1.

N. Balaskas, A. Ribeiro, J. Panovska, E. Dessaud, N. Sasai, K. Page, J. Briscoe, and V. Ribes. Gene regulatory logic for reading the sonic hedgehog signaling gradient in the vertebrate neural tube. *Cell*, 148:273–284, 2012.

J. M. Baldwin. A New Factor in Evolution. *The American Naturalist*, 1896. ISSN 0003-0147. doi: 10.1086/276408.

R. Baron, O. Lioubashevski, E. Katz, T. Niazov, and I. Willner. Elementary arithmetic operations by enzymes: A model for metabolic pathway based computing. *Angew. Chem. Int. Ed.*, 45:1572–1576, 2006.

U. S. Bhalla. Understanding complex signaling networks through models and metaphors. *Progress in Biophysics and Molecular Biology*, 81:45–65, 2003.

K. Bishop, G. Goudreau, and D. O'Leary. Regulation of area identity in the mammalian neocortex by Emx2 and Pax6. *Science*, 288(5464):344–349, 2000. doi: 10.1126/science.288.5464.344.

K. M. Bishop, J. L. R. Rubenstein, and D. D. M. O'Leary. Distinct actions of emx1, emx2, and pax6 in regulating the specification of areas in the developing neocortex. *Journal of Neuroscience*, 22(17):7627–7638, 2002. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.22-17-07627.2002. URL `http://www.jneurosci.org/content/22/17/7627`.

H. Bolouri. *Computational modeling of gene regulatory networks — A primer*. 2008. ISBN 9781848162228. doi: 10.1142/P567.

U. Borello, M. Madhavan, I. Vilinsky, A. Faedo, A. Pierani, J. Rubenstein, and K. Campbell. Sp8 and COUP-TF1 reciprocally regulate patterning and fgf signaling in cortical progenitors. *Cerebral Cortex*, 24(6):1409–1421, 2014. doi: 10.1093/cercor/bhs412.

D. Bray. Protein molecules as computational elements in living cells. *Nature*, 376 (6538):307–312, 1995.

D. Bray. Molecular networks: The top-down view. *Science*, 301:1864–1865, 2003.

G. Bronchti, P. Heil, R. Sadka, A. Hess, H. Scheich, and Z. Wollberg. Auditory activation of 'visual' cortical areas in the blind mole rat (Spalax ehrenbergi). *European Journal of Neuroscience*, 2002. ISSN 0953816X. doi: 10.1046/j.1460-9568.2002.02063.x.

N. E. Buchler, U. Gerland, and T. Hwa. On schemes of combinatorial transcription logic. *Proceedings of the National Academy of Sciences of the United States of America*, 100(9):5136–5141, 2003. ISSN 00278424. doi: 10.1073/pnas.0930314100.

J. Call and M. Tomasello. Does the chimpanzee have a theory of mind? 30 years later. *Trends in Cognitive Sciences*, 12(5):187–192, 2008. ISSN 13646613. doi: 10.1016/j.tics.2008.02.010.

S. B. Carroll. Evo-Devo and an Expanding Evolutionary Synthesis: A Genetic Theory of Morphological Evolution. *Cell*, 134(1):25–36, 2008. ISSN 00928674. doi: 10.1016/j.cell.2008.06.030.

F. Crick. Diffusion in embryogenesis. *Nature*, 1970. ISSN 00280836. doi: 10.1038/225420a0.

C. Darwin and A. Wallace. On the Tendency of Species to form Varieties; and on the Perpetuation of Varieties and Species by Natural Means of Selection. *Journal of the Proceedings of the Linnean Society of London. Zoology*, 1858. doi: 10.1111/j.1096-3642.1858.tb02500.x.

T. Deacon. The hierarchic logic of emergence: Untangling the interdependence of evolution and self-organization. *Evolution and Learning: The Baldwin effect reconsidered*, 2003.

D. C. Dennett. The Baldwin Effect: A Crane, Not a Skyhook. In *And Learning: The Baldwin Effect Reconsidered*. 2003.

W. Driever and C. Nüsslein-Volhard. A gradient of bicoid protein in Drosophila embryos. *Cell*, 1988. ISSN 00928674. doi: 10.1016/0092-8674(88)90182-1.

C. A. Dye, H. El Shawa, and K. J. Huffman. A Lifespan Analysis of Intraneocortical Connections and Gene Expression in the Mouse I. *Cerebral Cortex*, 21 (6):1311–1330, 11 2010a. ISSN 1047-3211. doi: 10.1093/cercor/bhq212. URL `https://doi.org/10.1093/cercor/bhq212`.

C. A. Dye, H. El Shawa, and K. J. Huffman. A Lifespan Analysis of Intraneocortical Connections and Gene Expression in the Mouse II. *Cerebral Cortex*, 21 (6):1331–1350, 11 2010b. ISSN 1047-3211. doi: 10.1093/cercor/bhq213. URL `https://doi.org/10.1093/cercor/bhq213`.

R. A. Fisher. XV.—The Correlation between Relatives on the Supposition of Mendelian Inheritance. *Transactions of the Royal Society of Edinburgh*, 1919. ISSN 00804568. doi: 10.1017/S0080456800012163.

R. A. Fisher. *The Genetical Theory of Natural Selection*. Oxford, Clarendon Press, 1930.

G. Frigerio, M. Burri, D. Bopp, S. Baumgartner, and M. Noll. Structure of the segmentation gene paired and the drosophila prd gene set as part of a gene network. *Cell*, 47:735–746, 1986.

T. Fukuchi-Shimogori and E. A. Grove. Neocortex patterning by the secreted signaling molecute FGF8. *Science*, 2001. ISSN 00368075. doi: 10.1126/science.1064252.

S. Gavrilets and J. Gravner. Percolation on the fitness hypercube and the evolution of reproductive isolation. *Journal of Theoretical Biology*, 1997. ISSN 00225193. doi: 10.1006/jtbi.1996.0242.

C. Gershenson. Guiding the self-organization of random Boolean networks. *Theory in Biosciences*, 2012. ISSN 14317613. doi: 10.1007/s12064-011-0144-x.

R. Gesztelyi, J. Zsuga, A. Kemeny-Beke, B. Varga, B. Juhasz, and A. Tosaki. The Hill equation and the origin of quantitative pharmacology, 2012. ISSN 00039519.

M. Gherardi and P. Rotondo. Measuring logic complexity can guide pattern discovery in empirical systems. *Complexity*, 2016. ISSN 10990526. doi: 10.1002/cplx.21819.

C. E. Giacomantonio and G. J. Goodhill. A boolean model of the gene regulatory network underlying mammalian cortical area development. *PLOS Computational Biology*, 6(9):1–13, 09 2010. doi: 10.1371/journal.pcbi.1000936. URL `https://doi.org/10.1371/journal.pcbi.1000936`.

N. K. Gordon and R. Gordon. The organelle of differentiation in embryos: The cell state splitter, 2016. ISSN 17424682.

S. J. Gould and N. Eldredge. Punctuated equilibria: The tempo and mode of evolution reconsidered. *Paleobiology*, 1977. ISSN 19385331. doi: 10.1017/S0094837300005224.

L. C. Greig, M. B. Woodworth, M. J. Galazo, H. Padmanabhan, and J. D. Macklis. Molecular logic of neocortical projection neuron specification, development and diversity, 2013. ISSN 1471003X.

E. A. Grove and T. Fukuchi-Shimogori. Generating the cerebral cortical area map. *Annu. Rev. Neurosci.*, 26:355–380, 2003.

C. C. Guet, M. B. Elowitz, W. Hsing, and S. Leibler. Combinatorial synthesis of genetic networks. *Science*, 296:1466–1470, 2002.

M. Gulisano, V. Broccoli, C. Pardini, and E. Boncinelli. Emx1 and emx2 show different patterns of expression during proliferation and differentiation of the developing cerebral cortex in the mouse. *Eur. J. Neurosci.*, 8:1037–1050, 1996.

J. B. S. Haldane. The Time of Action of Genes, and Its Bearing on some Evolutionary Problems. *The American Naturalist*, 1932. ISSN 0003-0147. doi: 10.1086/280406.

T. Hamasaki, A. Leingärtner, T. Ringstedt, and D. D. O'Leary. Emx2 regulates sizes and positioning of the primary sensory and motor areas in neocortex by direct specification of cortical progenitors. *Neuron*, 43(3):359–372, 2004. ISSN 0896-6273. doi: https://doi.org/10.1016/j.neuron.2004.07.016. URL `http://www.sciencedirect.com/science/article/pii/S0896627304004222`.

S. Harris, B. Sawhill, A. Wuensche, and S. Kauffman. A model of transcriptional regulatory networks based on biases in the observed regulation rules. *Complexity*, 7(4):23—40, 2002.

J. Hasty, D. McMillen, F. Isaacs, and J. J. Collins. Computational studies of gene regulatory networks: in numero molecular biology. *Nature reviews genetics*, 2: 268–279, 2001.

G. Hinton and S. Nolan. How learning can guide evolution. *Complex Systems*, 1: 495–502, 1987.

A. Hodgkinson and A. Eyre-Walker. Variation in the mutation rate across mammalian genomes. *Nature Reviews Genetics*, 12(11):756–766, 2011. doi: 10.1038/nrg3098.

K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

J. Huxley. *Evolution: The Modern Synthesis*. London: Allen Unwin, 1942.

I. Imayoshi, A. Isomura, Y. Harima, K. Kawaguchi, H. Kori, H. Miyachi, T. Fujiwara, F. Ishidate, and R. Kageyama. Oscillatory control of factors determining multipotency and fate in mouse neural progenitors. *Science*, 2013. ISSN 10959203. doi: 10.1126/science.1242366.

J. Jaeger and A. Crombach. Life's attractors understanding developmental systems through reverse engineering and in silico evolution. *Advances in Experimental Medicine and Biology*, 2012. ISSN 00652598. doi: 10.1007/978-1-4614-3567-9$_5$.

J. Jaeger and A. Martinez-Arias. Getting the Measure of Positional Information. *PLoS Biology*, 7(3):e1000081, 2009. doi: 10.1371/journal.pbio.1000081.

J. Jaeger and N. Monk. Bioattractors: dynamical systems theory and the evolution of regulatory processes. *The Journal of Physiology*, 2014. ISSN 00223751. doi: 10.1113/jphysiol.2014.272385.

C. Jung, P. Bandilla, M. Von Reutern, M. Schnepf, S. Rieder, U. Unnerstall, and U. Gaul. True equilibrium measurement of transcription factor-DNA binding affinities using automated polarization microscopy. *Nature Communications*, 9(1):1–11, 2018. ISSN 20411723. doi: 10.1038/s41467-018-03977-4. URL `http://dx.doi.org/10.1038/s41467-018-03977-4`.

M. Kaschube, M. Schnabel, S. Löwel, D. M. Coppola, L. E. White, and F. Wolf. Universality in the evolution of orientation columns in the visual cortex. *Science*, 2010. ISSN 00368075. doi: 10.1126/science.1194869.

R. Kast and P. Levitt. Precision in the development of neocortical architecture: From progenitors to cortical networks. *Prog Neurobiol.*, 175:77–95, 2019.

S. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22(3):437–467, 1969.

S. Kauffman. The ensemble approach to understand genetic regulatory networks. *Physica A: Statistical Mechanics and its Applications*, 340(4):733–740, 2004. doi: 10.1016/j.physa.2004.05.018.

S. A. Kauffman. *Origins of order: Self-organization and selection in evolution*. Oxford University Press, 1993.

M. Keynes and T. M. Cox. William Bateson, the rediscoverer of Mendel, 2008. ISSN 01410768.

M. C. King and A. C. Wilson. Evolution at two levels in humans and chimpanzees. *Science*, 1975. ISSN 00368075. doi: 10.1126/science.1090005.

J. Knabe, C. Nehaniv, M. Schilstra, and T. Quick. Evolving Biological Clocks using Genetic Regulatory Networks. *Artificial Life X : Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems*, 2006.

L. Krubitzer and J. Kaas. The evolution of the neocortex in mammals: How is phenotypic diversity generated?, 2005. ISSN 09594388.

L. Krubitzer and A. Seelke. Cortical evolution in mammals: The bane and beauty of phenotypic variability. *Proceedings of the National Academy of Sciences of the United States of America*, 109(SUPPL.1):10647–10654, 2012. doi: 10.1073/pnas.1201891109.

L. A. Krubitzer and T. J. Prescott. The Combinatorial Creature : Cortical Phenotypes within and across Lifetimes. *Trends in Neurosciences*, 41 (10):744–762, 2018. ISSN 0166-2236. doi: 10.1016/j.tins.2018.08.002. URL https://doi.org/10.1016/j.tins.2018.08.002.

J. Lewis. From signals to patterns: Space, time, and mathematics in developmental biology, 2008. ISSN 00368075.

R. A. Lewis, B. T. Wakimoto, R. E. Denelli, and T. C. Kaufman. Genetic analysis of the antennapedia gene complex (ant-C) and adjacent chromosomal regions of Drosophila melanogaster. II. Polytene chromosome segments 84A-84B1,2. *Genetics*, 1980. ISSN 00166731. doi: 10.1093/genetics/95.2.383.

F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang. The yeast cell-cycle network is robustly designed. *Proceedings of the National Academy of Sciences of the United States of America*, 101(14):4781 LP – 4786, apr 2004. doi: 10.1073/pnas.0305937101. URL http://www.pnas.org/content/101/14/4781.abstract.

H. D. Lipshitz. Follow the mRNA: A new model for Bicoid gradient formation, 2009. ISSN 14710072.

Q. Liu, N. D. Dwyer, and D. D. O'Leary. Differential expression of coup-tfi, chl1, and two novel genes in developing neocortex identified by differential display pcr. *J. Neurosci.*, 20:7682–7690, 2000.

J. H. Lui, D. V. Hansen, and A. R. Kriegstein. Development and evolution of the human neocortex, 2011. ISSN 00928674.

A. Mallamaci and A. Stoykova. Gene networks controlling early cerebral cortex areal-

ization. *European Journal of Neuroscience*, 23(4):847–856, 2006. doi: 10.1111/j.1460-9568.2006.04634.x.

M. Manuel, P. A. Georgala, C. B. Carr, S. Chanas, D. A. Kleinjan, B. Martynoga, J. O. Mason, M. Molinek, J. Pinson, T. Pratt, J. C. Quinn, T. I. Simpson, D. A. Tyas, V. van Heyningen, J. D. West, and D. J. Price. Controlled overexpression of pax6 in vivo negatively autoregulates the pax6 locus, causing cell-autonomous defects of late cortical progenitor proliferation with little effect on cortical arealization. *Development*, 134(3):545–555, 2007. ISSN 0950-1991. doi: 10.1242/dev.02764. URL `http://dev.biologists.org/content/134/3/545`.

W. McCulloch and W. Pitts. Logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

C. Métin and D. O. Frost. Visual responses of neurons in somatosensory cortex of hamsters with experimentally induced retinal projections to somatosensory thalamus. *Proceedings of the National Academy of Sciences of the United States of America*, 1989. ISSN 0027-8424. doi: 10.1073/pnas.86.1.357.

M. Minsky and S. Papert. *Perceptrons: An introduction to computational geometry*. MIT Press, 1969.

Y. Nakagawa, D. D. M. O'Leary, and J. E. Johnson. Graded and areal expression patterns of regulatory genes and cadherins in embryonic neocortex independent of thalamocortical input. *Journal of Neuroscience*, 1999. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.19-24-10877.1999.

C. L. Nehaniv. Self-replication, evolvability and asynchronicity in stochastic worlds. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial*

Intelligence and Lecture Notes in Bioinformatics), 2005. ISBN 3540294988. doi: $10.1007/11571155_13$.

A. S. Nord, K. Pattabiraman, A. Visel, and J. L. Rubenstein. Genomic Perspectives of Transcriptional Regulation in Forebrain Development. *Neuron*, 85 (1):27–47, 2015. ISSN 10974199. doi: 10.1016/j.neuron.2014.11.011. URL `http://dx.doi.org/10.1016/j.neuron.2014.11.011`.

D. O'Leary and Y. Nakagawa. Patterning centers, regulatory genes and extrinsic mechanisms controlling arealization of the neocortex. *Current Opinion in Neurobiology*, 12(1):14–25, 2002. doi: 10.1016/S0959-4388(02)00285-4. cited By 218.

D. D. O'Leary. Do cortical areas emerge from a protocortex?, 1989. ISSN 01662236.

K. Pattabiraman, O. Golonzhka, S. Lindtner, A. S. Nord, L. Taher, R. Hoch, S. N. Silberberg, D. Zhang, B. Chen, and H. Zeng. Transcriptional regulation of enhancers active in protodomains of the developing cerebral cortex. *Neuron*, 82(5):989–1003, 2014. doi: 10.1016/j.neuron.2014.04.014.Transcriptional.

U. Paul, V. Kaufman, and B. Drossel. Properties of attractors of canalyzing random Boolean networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 2006. ISSN 15393755. doi: 10.1103/PhysRevE.73.026118.

I. Peter and E. Davidson. Evolution of Gene Regulatory Networks Controlling Body Plan Development. *Cell*, 144(6):970–985, mar 2011. ISSN 0092-8674. doi: 10.1016/j.cell.2011.02.017.

M. Pigliucci and G. B. Muller. *Evolution: the Extended Synthesis*. MIT Press, 2010.

F. J. Pineda. Generalization of back-propagation to recurrent neural networks. *Phys. Rev. Lett.*, 59:2229–2232, Nov 1987. doi: 10.1103/PhysRevLett.59.2229. URL https://link.aps.org/doi/10.1103/PhysRevLett.59.2229.

F. J. Pineda. Recurrent Backpropagation and the Dynamical Approach to Adaptive Neural Computation. *Neural Computation*, 1(2):161–172, 06 1989. ISSN 0899-7667. doi: 10.1162/neco.1989.1.2.161. URL https://doi.org/10.1162/neco.1989.1.2.161.

K. Prüfer, F. Racimo, N. Patterson, F. Jay, S. Sankararaman, S. Sawyer, A. Heinze, G. Renaud, P. H. Sudmant, C. De Filippo, H. Li, S. Mallick, M. Dannemann, Q. Fu, M. Kircher, M. Kuhlwilm, M. Lachmann, M. Meyer, M. Ongyerth, M. Siebauer, C. Theunert, A. Tandon, P. Moorjani, J. Pickrell, J. C. Mullikin, S. H. Vohr, R. E. Green, I. Hellmann, P. L. Johnson, H. Blanche, H. Cann, J. O. Kitzman, J. Shendure, E. E. Eichler, E. S. Lein, T. E. Bakken, L. V. Golovanova, V. B. Doronichev, M. V. Shunkov, A. P. Derevianko, B. Viola, M. Slatkin, D. Reich, J. Kelso, and S. Pääbo. The complete genome sequence of a Neanderthal from the Altai Mountains. *Nature*, 2014. ISSN 00280836. doi: 10.1038/nature12886.

P. Rakic. Specification of cerebral cortical areas. *Science*, 241(4862):170–176, 1988. ISSN 00368075. doi: 10.1126/science.3291116.

P. Rakic. Evolution of the neocortex: Perspective from developmental biology. *Nature Reviews Neuroscience*, 10(10):724–735, 2009. doi: 10.1038/nrn2719.

P. Rakic, A. E. Ayoub, J. J. Breunig, and M. H. Dominguez. Decision by division: making cortical maps. *Trends in Neurosciences*, 32(5):291–301, 2009.

K. W. Rogers and A. F. Schier. Morphogen Gradients: From Generation to

Interpretation. *Annual Review of Cell and Developmental Biology*, 27(1):377–407, 2011. ISSN 1081-0706. doi: 10.1146/annurev-cellbio-092910-154148. URL http://www.annualreviews.org/doi/10.1146/annurev-cellbio-092910-154148.

F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 1958.

D. Rumelhart, G. Hinton, and R. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

A. Saghatelian, N. Volcker, K. Guckian, V. Lin, and M. Ghadiri. Dna-based photonic logic gates: And, nand, and inhibit. *J. Am. Chem. Soc.*, 125:346–347, 2003.

S. Sahara, Y. Kawakami, J. Belmonte, and D. O'Leary. Sp8 exhibits reciprocal induction with Fgf8 but has an opposing effect on anterior-posterior cortical area patterning. *Neural Development*, 2(1), 2007. doi: 10.1186/1749-8104-2-10.

I. Salazar-Ciudad. Developmental constraints vs. variational properties: How pattern formation can help to understand evolution and development, 2006. ISSN 15525007.

S. N. Sansom and F. J. Livesey. Gradients in the brain: the control of the development of form and function in the cerebral cortex., 2009. ISSN 19430264.

S. N. Sansom, J. M. Hébert, U. Thammongkol, J. Smith, G. Nisbet, M. A. Surani, S. K. McConnell, and F. J. Livesey. Genomic characterisation of a fgf-regulated gradient-based neocortical protomap. *Development*, 132(17):3947–3961, 2005. ISSN 0950-1991. doi: 10.1242/dev.01968.

B. L. Schlaggar and D. D. O'Leary. Potential of visual cortex to develop an array of functional units unique to somatosensory cortex. *Science*, 1991. ISSN 00368075. doi: 10.1126/science.2047863.

T. Shimogori and E. A. Grove. Fibroblast growth factor 8 regulates neocortical guidance of area-specific thalamic innervation. *Journal of Neuroscience*, 25(28): 6550–6560, 2005. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.0453-05.2005. URL http://www.jneurosci.org/content/25/28/6550.

H. Shimojo, T. Ohtsuka, and R. Kageyama. Oscillations in Notch Signaling Regulate Maintenance of Neural Progenitors. *Neuron*, 2008. ISSN 08966273. doi: 10.1016/j.neuron.2008.02.014.

A. M. Sousa, K. A. Meyer, G. Santpere, F. O. Gulden, and N. Sestan. Evolution of the Human Nervous System Function, Structure, and Development. *Cell*, 170(2): 226–247, 2017. ISSN 10974172. doi: 10.1016/j.cell.2017.06.036.

M. I. Stefan and N. Le Novère. Cooperative Binding. *PLOS Computational Biology*, 9(6):1–6, 2013. doi: 10.1371/journal.pcbi.1003106. URL https://doi.org/10.1371/journal.pcbi.1003106.

P. Stirling and S. Laughlin. *Principles of Neural Design*. MIT Press, 2015.

M. Sur and J. L. Rubenstein. Patterning and plasticity of the cerebral cortex, 2005. ISSN 00368075.

M. Sur, P. E. Garraghty, and A. W. Roe. Experimentally induced visual projections into auditory thalamus and cortex. *Science*, 1988. ISSN 00368075. doi: 10.1126/science.2462279.

S. C. Suzuki, T. Inoue, Y. Kimura, T. Tanaka, and M. Takeichi. Neuronal circuits are subdivided by differential expression of type-II classic cadherins in postnatal mouse brains. *Molecular and Cellular Neurosciences*, 1997. ISSN 10447431. doi: 10.1006/mcne.1997.0626.

A. Szejka and B. Drossel. Evolution of canalizing Boolean networks. *European Physical Journal B*, 2007. ISSN 14346028. doi: 10.1140/epjb/e2007-00135-2.

R. Toyoda, S. Assimacopoulos, J. Wilcoxon, A. Taylor, P. Feldman, A. Suzuki-Hirano, T. Shimogori, and E. A. Grove. FGF8 acts as a classic diffusible morphogen to pattern the neocortex. *Development*, 137(20):3439–3448, 2010. ISSN 09501991. doi: 10.1242/dev.055392.

A. Turing. The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 1952. ISSN 2054-0280. doi: 10.1098/rstb.1952.0012.

H. Van Der Loos and T. A. Woolsey. Somatosensory cortex: Structural alterations following early injury to sense organs. *Science*, 1973. ISSN 00368075. doi: 10.1126/science.179.4071.395.

B. Verd, N. A. Monk, and J. Jaeger. Modularity, criticality, and evolvability of a developmental gene regulatory network. *eLife*, 2019. doi: 10.7554/elife.42832.

J. Vohradsky. Neural network model of gene expression. *The FASEB journal*, 15: 846–854, 2001.

R. e. a. Waclaw. The zinc finger transcription factor sp8 regulates the generation and diversity of olfactory bulb interneurons. *Neuron*, 49:503–516, 2006.

C. Walther and P. Gruss. Pax-6, a murine paired box gene, is expressed in the developing cns. *Development*, 113:1435–1449, 1991.

J. D. Watson and F. H. Crick. Molecular structure of nucleic acids: A structure for deoxyribose nucleic acid. *Nature*, 1953. ISSN 00280836. doi: 10.1038/171737a0.

B. Weber and D. Depew. *Evolution and learning: The Baldwin effect reconsidered*. MIT Press, 2003.

S. P. Wilson and J. A. Bednar. What, if anything, are topological maps for? *Developmental Neurobiology*, 2015. ISSN 1932846X. doi: 10.1002/dneu.22281.

S. P. Wilson, S. S. James, D. J. Whiteley, and L. A. Krubitzer. Limit cycle dynamics can guide the evolution of gene regulatory networks towards point attractors. *Scientific Reports*, 2019. ISSN 20452322. doi: 10.1038/s41598-019-53251-w.

F. Wolf. Symmetry, multistability, and long-range interactions in brain development. *Physical Review Letters*, 2005. ISSN 00319007. doi: 10.1103/PhysRevLett.95.208701.

L. Wolpert. Positional information and the spatial pattern of cellular differentiation. *Journal of Theoretical Biology*, 25, 1969. ISSN 10958541. doi: 10.1016/S0022-5193(69)80016-0.

L. Wolpert. One hundred years of positional information. *Trends in Genetics*, 12(9): 359–364, 1996. doi: 10.1016/S0168-9525(96)80019-9.

G. A. Wray, M. W. Hahn, E. Abouheif, J. P. Balhoff, M. Pizer, M. V. Rockman, and L. A. Romano. Review Article The Evolution of Transcriptional Regulation in Eukaryotes. *Molecular Biology and Evolution, Vol. 20, No. 9*, pages 1377–1419, 2003. doi: 10.1093/molbev/msg140.

S. Wright. The roles of mutation, inbreeding, crossbreeding and selection in evolution. *Sixth International Congress on Genetics*, 1932.

A. R. Ypsilanti and J. L. Rubenstein. Transcriptional and epigenetic mechanisms of early cortical development –an examination of how pax6 coordinates cortical development. *The Journal of Comparative Neurology*, 524(3):609–629, 02 2016.

C. Zhou, S. Y. Tsai, and M. J. Tsai. Coup-tfi: an intrinsic factor for early regionalization of the neocortex. *Genes Dev.*, 15:2054–2059, 2001.

K. Zilles and K. Amunts. Centenary of Brodmann's map conception and fate, 2010. ISSN 1471003X.