LU TP 18-10
May 2018

Modelling conversion of adult skin cells to neurons

Viktor Drugge

Department of Astronomy and Theoretical Physics, Lund University

Master thesis supervised by Victor Olariu and Carsten Peterson

Abstract

Scientists are now able to directly convert one somatic cell type into another using a procedure known as direct lineage reprogramming or transdifferentiation. In this procedure, transcription factors which are important for initiating a rewriting of the gene expressions are introduced in the cell. One specific type of reprogramming involves generating dopamine producing neurons from human adult fibroblast skin cells. The transdifferentiation procedure in human cells has proven challenging. So far, conversion schemes are not able to generate satisfactory levels of mature neurons. However, experimental efforts are made to overcome this. Succeeding in generating a high yield conversion scheme would open up new pathways for medical treatments and disease modeling of diseases such as Parkinson's disease.

In this thesis, we study a model built in silico for a gene circuit proven to be important in the transdifferentiation from human adult fibroblast cells to neurons. Using experimental time series of gene expression obtained from a recently found high-yield neural conversion scheme, the model is capable to capture the experimental data dynamics. The system exhibits at least two attractors: one representing a neuronal state, and the other a non-neuronal state. A stochastic simulation was conducted for identifying strategies leading to high-yield neural conversion. The aim of the model presented here is to improve our understanding of the underlying dynamics, which may lead to a high yield neural conversion scheme applicable in vitro and in vivo.

Populärvetenskaplig sammanfattning

Gener fungerar som instruktioner för vad cellen skall producera. Det är aktiveringen och hämningen av gener som ger upphov till många olika typer av celler i människans kropp som till exempel hudceller, blodceller och nervceller. Dessa celler samarbetar med varandra för att tillsammans skapa den flercelliga organismen: människan. Det finns två huvudkategorier av celler i kroppen, somatiska celler samt könsceller. Nervceller och hudceller tillhör alltså gruppen somatiska celler.

Skadade somatiska celler kan leda till sjukdomar som till exempel Parkinsons sjukdom. Denna sjukdom förknippas med en nedsatt produktion av signalsubstansen dopamin. Dopamin produceras av nervceller i hjärnan och är viktig för till exempel motoriken i kroppen. Ett alternativ är att byta ut de skadade nervcellerna mot friska celler, men detta medför etiska dilemman då vävnaden som transplanteras tas från aborterade foster.

Under senare år har forskning lett till att en ny typ av teknik utvecklats som involverar omprogrammering av patientens egna celler. Genom att injicera olika ämnen i cellen är det möjligt att ändra vilka gener som aktiveras och hämmas. På så vis är det möjligt att ändra celltypen för somatiska celler. Till en början skedde omprogrammeringen via ett stamcellsstadium, men denna teknik har visat sig ha en förhöjd risk av tumörbildning. Istället används direktomvandling, en omprogrammeringsstrategi där en somatisk celltyp direkt övergår till en annan utan det mellanliggande stadiet. Med denna metod är det möjligt att till exempel generera nervceller direkt från hudceller. Än så länge är andelen celler som omvandlas relativt låg hos vuxna individer. Detta är en svårighet som behöver lösas för att metoden skall bli användbar i praktiken.

Vi har gjort en närmare studie av genregleringen i direktomvandling från hudceller till nervceller hos vuxna människor. Samspelet mellan generna skapar ett avancerat nätverk som beskriver omvandlingen på en molekylär nivå. Genom att kombinera etablerade interaktioner med ett antal hypotetiska interaktioner har vi lyckats hitta ett nätverk som beskriver experimentell data i datorsimuleringar. Studier av detta nätverk kan ge en inblick i de underliggande processerna som styr direktomvandling mellan hud- och nervceller. En vidareutveckling av modellen så att dopaminproducerande nervceller inkluderas skulle kunna leda till en utökad förståelse av direktomvandling från hud- till nervcell. Detta kan i slutändan ge effektivare omvandlingsstrategier vilket har stor potential för medicinska metoder i kampen mot till exempel Parkinsons sjukdom.

Acknowledgements

A big thank you to my supervisor Victor Olariu for providing feedback and steering me through this master thesis. Another big thank you to Adriaan Merlevede for fruitful discussions and constructive critique along the way. Many thanks to Carsten Peterson for valuable feedback during my presentation practice session. I also want to thank my reviewers Patrik Edén and Tobias Ambjörnsson for helpful feedback on the thesis. Finally, a big thank you to my family and friends for supporting and cheering me on during this time.

Contents

List of Figures

List of Tables

List of Abbreviations

Notation	**Description**
Ascl1	A neuronal gene, often used in transdifferentiation between HAF and neuron.
Brn2	A neuronal gene, involved in the maturation of reprogrammed neurons.
HAF	Human Adult Fibroblast, a cell type that build connective tissue between organs in the body.
iPS	Induced Pluripotent Stem cell.
miR	microRNA, small non-coding RNA molecules that regulate gene expression.
miR-124	A neuronal miR, associated with silencing non-neuronal gene expression.
miR-9	A neuronal miR, associated with silencing non-neuronal gene expression.
mRNA	messenger RNA, transcribed from the gene sequence on the DNA.
nPTB	Neural PTB, a neuronal gene that is upregulated in neuronal cells.
PTB	A non-neuronal gene, upregulated in non-neuronal cells.
REST	A non-neuronal gene that together with other components build the REST complex.
RESTc	REST complex, a protein complex that represses many neuronal genes. It is built from REST protein and components it recruits.
RESTi	REST inhibition, a factor that represses the gene expression of REST.
SCP1	A non-neuronal gene, a component of RESTc.
shREST	Short Hairpin REST, equivalent to RESTi.

1 Background

The body is constructed by a vast amount of different cell types that range from skin cells to nerve cells. It is the gene expression that define these cells. As a result, skin cells and nerve cells have different functions while possessing the same DNA. Collectively, the cells that build a multicellular organism constitute two major categories: somatic cells and reproductive cells. The group of somatic cell types is the largest.

In humans, pathological somatic cells lead to diseases such as Parkinson's disease, which is characterized by ceased production of dopamine by neurons in the brain[1]. A relatively new type of treatment known as regenerative treatment is used to rejuvenate cells, tissues and organs of the human body. By replacing unhealthy cells the treatment aim to establish normal function of the organs.

For nerve damage and other types of cell related illnesses, cell therapy, a branch of regenerative treatment, is used. In this medical treatment, cells are transplanted to the patient with the intention of restoring damaged nerves. From the end of the twentieth century there have been clinical trials using cell therapy performed on Parkinson's disease patients[2]. Results were promising with some patients having long-lasting and pronounced effects after a transplantation, allowing for medication withdrawal. However, the treatment is not free from ethical controversies as the tissue being transplanted is taken from fetuses[2,3].

A new branch of medical research, free of ethical disputes, involves transplanting the patient's own cells. In this medical treatment, somatic cells are reprogrammed to stem cell-like states using a cocktail of gene promoting or silencing factors. The factors are transducted, inserted using a viral vector or virus, inside the somatic cells in vitro. As the cells begin to produce messenger RNAs (mRNAs) for stem cells and repress mRNAs specific for the current somatic cell type, a conversion occurs. The cells are converted to a stem cell-like state ready for differentiation to the desired somatic type. The differentiated cells are then transplanted into the patient.

Recently, somatic cells have been reprogrammed to embryonic-like states using transcription factors to revert the differentiation of the cell. In 2006, Takahashi and Yamanaka[4] managed to reprogram mouse somatic cells to a pluripotent state denoted induced pluripotent stem (iPS) cells. Later on, in 2007, Takahashi et al.[5] and Yu et al.[6] also managed to reprogram human adult fibroblast (HAF) to iPS cells using a set of four transcription factors. These mouse and human iPS cells exhibited similar characteristics such as morphology and proliferation found in their embryonic stem cell counterparts. As mentioned in the report by Takahashi et al. their results suggest that a fundamental transcriptional network governs the pluripotency in human and mouse cells, but external factors maintaining the pluripotency are different for the species.

An issue with iPS cells that needs to be addressed before starting clinical trials on humans, is the development of tumors[7]. This has led to research on a new technique known as direct lineage conversion[8]. Direct lineage conversion, also known as transdifferentiation, is a technique that forces a somatic cell directly to another somatic cell type without the intermediate pluripotent state. The advantage of using this method is that the unstable pluripotent state is avoided, reducing the tumorigenicity of the cells. In a similar fashion to iPS cell generation, a cocktail of gene regulating factors are transducted into the cell.

Research on the transdifferentiation from fibroblast to neuron suggest there exists key regulatory genes important for the reprogramming. Here a brief introduction to the genes considered in this study is given. Previously, a cocktail using transcription factors Ascl1, Brn2, and Myt1l, has been used to achieve direct lineage conversion from mouse embryonic fibroblast cells to induced neuron cells[9]. Ascl1 was shown to be the driving mechanism of the neuronal reprogramming[10]. This work inspired research on human cells, converting fibroblast cells to induced neuron cells[11–15]. Interestingly, using the previous cocktail for mouse embryonic fibroblast cells resulted in immature induced neuron cells in humans[13,14]. Despite this, all studies reported an optimal cocktail for reprogramming using one or more of the three transcription factors used on mouse, in combination with other factors.

Continuing the discussion on regulatory genes, Yoo et al.[12] reported that human fibroblasts could be reprogrammed using microRNA (miR) 124 and 9 combined with Ascl1 and other transcription factors. These miRs are small molecules that help regulate gene expression. Two important genes regulated by miR-124 and miR-9 is SCP1[16] and REST[17], respectively. The REST factor represses a large set of neuronal genes by binding to their respective DNA site and preventing their gene expression[18,19]. At the binding site it forms a complex (RESTc) by recruiting co-factors[19]. Among the co-factors recruited is SCP1. This factor has shown to be influential in neuronal gene repression of non-neuronal cells[20]. Another important component that is repressed by miR-124, is the PTB protein[21]. The PTB protein is upregulated in non-neuronal cells and downregulated in neuronal cells. An anti-correlated gene associated with PTB down- and upregulation is neural PTB (nPTB). One reason for this behavior is that PTB represses nPTB[21,22].

The mentioned genes have recently been combined into two regulatory loops. In a 2013 paper, the first regulatory loop was found that induced transdifferentiation from mouse embryonic fibroblast cells to neurons by knockdown of PTB[23]. Later, a study on HAF cells showed that in humans, a second regulatory loop is important for the maturation process[24]. The two regulatory loops consist of Brn2, miR-124, miR-9, RESTc, PTB, and nPTB; genes that, as previously mentioned, are important for fibroblast to neuron transdifferentiation in human.

Being able to generate induced neuron cells from HAF cells, using direct lineage reprogramming pose an exciting new method for disease modeling and medical applications of neurodegenerative diseases. An issue using this method is the relatively low efficiency of reprogramming[14]. However, recently our collaborators Malin Parmar and Janelle Drouin-Ouellet at Lund university managed to improve the methods efficiency to relatively high yields[25]. They found that REST acts as a reprogramming barrier that prevents neuronal genes expression, in line with previous studies[23,24].

To gain further knowledge of the underlying mechanisms and to potentially improve the conversion scheme, we have studied here the neural conversion in silico. An advantage of in silico approaches, compared to in vitro or in vivo approaches, is the short time scale required. Here, testing dynamics of the transdifferentiation is relatively effortless and performed in a couple of days. Merging the regulatory loops mentioned above and incorporating the cocktail found by our collaborators, we have tested different regulatory gene network topologies. Associated with each topology is a set of rate equations with unknown parameters that describe how each gene expression evolves in time in a deterministic way. Provided with experimental time series data for Ascl1, Brn2, miR-124, miR-9, REST, SCP1, PTB, and nPTB, from our experimental collaborators, we found an optimal set of parameter values for each topology using two global search methods, simulated annealing and genetic algorithm. The best performing regulatory gene network is able to capture the main features of experimental time series data. The winning topology is also used to simulate an ensemble of 100 cells stochastically. We found that perturbing a component of RESTc resulted in cells either converting to a neuronal-like state or a non-neuronal like state. Thus, the model presented in this thesis might prove useful in furthering our knowledge of the underlying processes that guide HAF to neuron transdifferentiation in humans.

1.1 Model

In this section the network and its components will be discussed in more detail, providing an overview of the interactions between them. The goal is to introduce the nodes in the network, their interactions, and how we can build a network out of these. The section is concluded with a figure illustrating a basic regulatory gene network built from the interactions presented.

1.1.1 Two regulatory loops

In 2013, Xue et al.[23] were able to show that several cell types, for example human embryonic stem cells, differentiated into neuron-like states when a protein known as PTB was repressed. The conversion was governed by a network consisting of PTB, microRNA miR-124, and REST complex (RESTc), see figure 1.

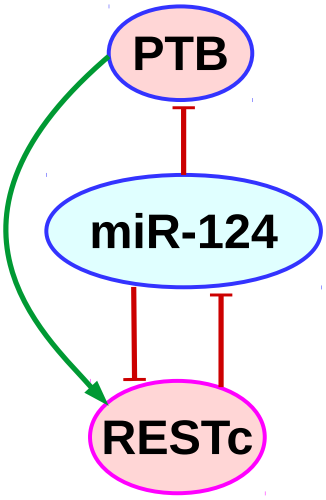


Figure 1: The first regulatory gene loop, important for the transdifferentiation process to a human neuron cell.

Here is a brief description of the genes in the network. PTB is a non-neuronal protein that functions as a negative regulator of neuronal genes in non-neuronal cells[22]. RESTc consists of the REST protein and the components recruited by this factor[19]. It acts as a reprogramming barrier, repressing several neuronal genes in non-neuronal cells[19,23,25]. This complex is built from REST and the components recruited by this protein at the gene site[18]. One such component is SCP1, a REST co-factor.

Yeo et al.[20] showed that SCP1 is being recruited by REST to serve an important role in the repression of neuronal genes in non-neuronal cells. By repressing SCP1 together with REST activity Yeo et al. were able to further enhance the neuronal conversion rate. miR-124 is a microRNA, derived from non-coding sequences (not translated into protein) of mRNA[26–28]. MicroRNA, once transcribed, regulates gene expression by binding to mRNA post-transcriptionally, preventing translation of the targeted gene. Thus, microRNAs are important regulators of gene expression.

Here follows a short description of the interactions we included in this study. The three mentioned nodes form a regulatory loop in which an increased regulation of PTB and RESTc translates to high repression of neuronal genes. Converting to a neuron state is accomplished by repressing both PTB and RESTc. As shown in Xue et al., both miR-124 and PTB have targeting sites on SCP1. Knockdown of PTB enhanced suppression of SCP1 and overexpression of miR-124 had a similar effect. We model RESTc as composed of two parts, REST and SCP1. This choice is based on studies[20,23], and that for our purposes the other RESTc co-factors appear less influential. In our model, PTB activates SCP1[23], miR-124 represses SCP1[16] and PTB[21], while REST complex represses miR-124[23], see figure 1.

In 2016, Xue et al.[24] reported that deactivating neuronal repression through the above mentioned regulatory loop on HAF cells produced immature neurons. They announced to have found a secondary loop, important for neuron maturation in human. This loop contains neural PTB (nPTB), the transcription factor Brn2, and the microRNA miR-9, see figure 2. Note that Ascl1 denote a combined node of Ascl1 and Brn2 which will be explained shortly. This regulatory loop is activated by repressing nPTB once transdifferentiation to neuron-state has occurred. The sequential process is important for producing functional neurons in humans[24].

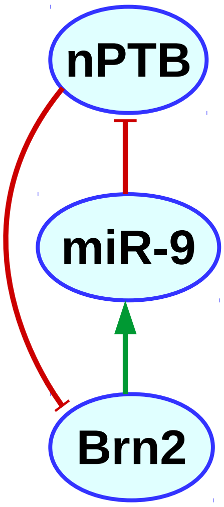


Figure 2: The second regulatory gene loop, important for the maturation process of a human neuron cell.

Here a short description of the reported genes is given. nPTB is similar in structure to its non-neuronal paralog PTB[22]. PTB is expressed in non-neuronal cells and nPTB is expressed in neuronal cells. Brn2 is a transcription factor used for neuronal conversion and important for the maturation of converted cells[9,13,24,25]. It is upregulated in maturing neurons and repressed in non-neuronal cells. MicroRNA miR-9 has, in similar fashion to miR-124, an important role in gene regulation[29]. Xue et al. found that both miR-124 and miR-9 serve a role of repressing nPTB in maturing neurons, however miR-124 targets nPTB relatively weakly in comparison to PTB[21,24].

Incorporating our collaborators reprogramming cocktail required the transcription factor Ascl1 to be introduced. Ascl1, similar to Brn2, is upregulated in the neuronal state[30].

In mouse cells neuronal reprogramming is driven by Ascl1, which recruits Brn2 to many of its binding sites[10,31]. It has been shown that overexpression of Ascl1 alone could induce human fibroblasts to cells with neuron-like phenotype[11]. In combination with the non-neuronal gene repressors miR-124 and miR-9, Ascl1 helps human neuronal reprogramming[12]. For these reasons, Ascl1 and Brn2 can be considered as a single node that drives neuronal transdifferentiation. In our models we combine Ascl1 and Brn2 into a single node that is denoted Ascl1.

The three mentioned nodes form the second regulatory loop, which is important for the maturation of immature neurons in humans. This subsequent paragraph explains the interactions incorporated in our model. In the second loop overexpression of Brn2 triggers the expression of neuronal genes by upregulating microRNAs 124 and 9.[24] In our model this is represented by overexpression of Ascl1. Xue et al.[24] reported that nPTB represses Brn2 since knockdown of PTB had small effects on Brn2 levels, while sequential knockdown of PTB and nPTB induced Brn2. Xue et al. also reported that Brn2 targets both miR-124 and miR-9. These interactions were included in our network as well. As mentioned above, microRNA miR-9 represses nPTB[24], also included in this study.

In summary, transdifferentiation from a fibroblast state to a neuron state is controlled by a regulatory loop consisting of PTB, miR-124, and RESTc. Activating conversion from a non-neuronal state to a neuronal state is achieved by repressing PTB and RESTc. Together, these genes create the first regulatory loop, summarized below.

- First regulatory loop
  - miR-124 represses PTB, and RESTc through SCP1
  - PTB activates RESTc through SCP1
  - RESTc represses miR-124

From the immature neuronal state the cell is matured using a second regulatory loop. This loop consists of nPTB, miR-9, and Ascl1 and Brn2. Together, these genes create the second regulatory loop, summarized below.

- Second regulatory loop
  - nPTB represses Brn2
  - miR-9 represses nPTB
  - Ascl1 activates miR-9

These genes constitute the basic nodes in our regulatory gene network.

1.1.2 Merging the two regulatory loops

The two regulatory loops in section 1.1.1 build the regulatory gene network. To gain further insight on the transdifferentiation scheme we model these two loops in silico. The loops are intertwined, and not independent of each other[24]. Therefore the gene expression of one node is affecting the gene expression of the other nodes. As a consequence they have to be merged.

Here the interactions that connect the two regulatory gene loops are described. In similar fashion to a previous study, the regulatory gene network is built from experimentally determined interactions[32]. Studies have shown that PTB is repressing nPTB[33,34]. In non-neuronal cells PTB is expressed in higher concentrations, and in neuronal cells nPTB is expressed in higher concentrations. Downregulation of PTB has a positive effect on nPTB expression which makes a connection between the two regulatory loops. As mentioned in the previous section, Brn2 activates both miR-124 and miR-9. This combined with the fact that Ascl1 could induce transdifferentiation in combination with miR-124 and miR-9 suggest that the Ascl1 node in our model activates both microRNAs. In addition, RESTc has targeting sites on Brn2 and Ascl1, as well as on miR-124 and miR-9.[23] Repression of RESTc in PTB downregulated cells resulted in induced expression of these four components. Since RESTc is associated with neuronal gene repression, and Ascl1, Brn2, miR-124, and miR-9, is associated with neuronal gene promotion, RESTc is likely a repressor of these 4 nodes. These interactions, together with those described in section 1.1.1, connect the two regulatory loops, forming the basic structure of the regulatory gene network. An example of a network topology is shown in figure 3. This network is based on previously discussed connections.

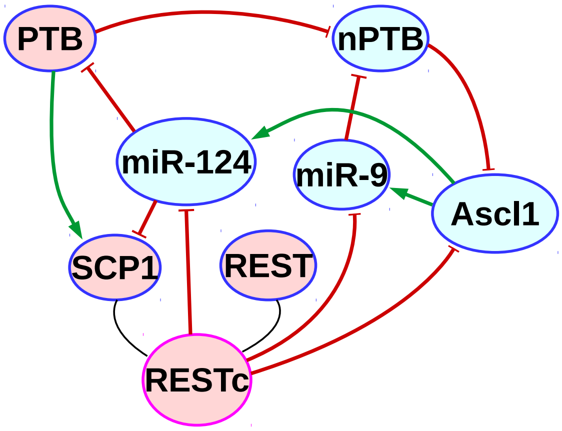


Figure 3: Basic topology of the regulatory gene network. The connections made between nodes constitute experimentally determined interactions. Note that Ascl1 and Brn2 are characterized by a single node denoted Ascl1. The different colour signals is explained in figure 7 on page 28.

2 Method

A theoretical study of a regulatory system and its evolution in time require mathematical definitions and principles to properly be understood. In this section, the aim is to define the methodologies used as well as providing a brief description of them.

2.1 Mathematical formalism

2.1.1 Gene interactions

The regulation of gene expression can be described mathematically in several different ways. Two common principles for describing gene expression in a regulatory gene network is the Hill-, respectively Shea and Ackers formalism. The Hill formalism constitutes an approach centered around the Hill equation[35]. This equation is a generalization of the Michaelis-Menten equation[36], a model of enzyme kinetics when a single substrate can bind to the enzyme. The Hill equation generalizes this to include n number of substrates, considering cooperativity in binding to the enzyme. The second formalism is known as the Shea and Ackers formalism, based on a statistical physics approach[37]. In this work, we use the Shea and Ackers formalism.

A gene denotes a sequence of nucleotides, the building blocks of DNA. In total, there exist four building blocks, A, T, C, and G. Forming hydrogen bonds with each other, these nucleotides create long double-stranded sequences, the DNA[38]. The DNA holds the genetic information that is used to build an organism. Gene expression is the rendering of this genetic information to important machines: proteins. This process can be divided into two main categories: transcription and translation[39].

Formulating transcription

Transcription of a gene is performed by the enzyme RNA polymerase[40,41]. The initiation of transcription starts with RNA polymerase binding to the site of the gene. In order to bind with this site, factors known as transcription factors are required. In eukaryotes, there are two main categories of transcription factors, general and specific[40]. The general transcription factors are required for the transcription of any gene, as they recruit and help RNA polymerase bind to the DNA. The specific transcription factors bind to specific nucleotide sequences, either facilitating the assembly of general transcription factors and RNA polymerase, or blocking this assembly. Thus, specific transcription factors either promote or repress the transcription of a gene. Once RNA polymerase is bound to the DNA, it may start walking along it, synthesizing mRNA.

The site where the RNA polymerase binds to the DNA is denoted the promoter. The promoter can be in various states s depending on the presence of RNA polymerase (RNAp) and transcription factors (TF), see figure 4. For a state s_m, let ℓ denote the number of transcription factors able to bind to the promoter, $i_{s_m,\ell}$ whether the transcription factor is bound or not, and j_{s_m} whether the RNAp is bound or not. The combination of all states form a partition function, $Z(s)$, of the promoter site, as shown below.

$$Z = \sum_{s_1 \ldots s_m \ldots s_n} \prod_{\ell} [\mathrm{TF}_\ell]^{i_{s_m,\ell}} [\mathrm{RNAp}]^{j_{s_m}} e^{-\frac{\Delta G(s_m)}{k_B T}}. \tag{1}$$

Here, the sum is over all possible states s and the product over all transcription factors able to bind on the promoter site. Each state s_m has a set of TFs and RNAp that are either bound or unbound. Note that the square brackets represent the concentration of the respective term. A TF that is bound to the promoter site have $i_{s_m,\ell}$ equal to an integer larger than 0, and a RNAp that is bound to the promoter site have j_{s_m} equal to an integer larger than 0. An unbound TF or RNAp have $i_{s_m,\ell}$ or j_{s_m} equal to 0. In the Shea and Ackers formalism, $i_{s_m,\ell}$ and j_{s_m} represents the number of proteins bound of each species to the promoter site. The exponential term depends on the Gibbs free energy, $\Delta G(s_m)$, of the state. This factor symbolizes the difference in free energy between the unbound state and the bound state s_m. Note that $\Delta G(s_m) = 0$ when s_m is the state with no TFs or RNAp bound. The exponential term also depends on the Boltzmann constant k_B, and the absolute temperature T. This exponential term is often denoted the binding affinity parameter, $K = e^{\frac{\Delta G(s_m)}{k_B T}}$. For reasons that will be discussed in section 2.1.1, we absorb the RNAp concentration in the binding affinity parameter, which is treated as an unknown parameter in our model.

The total partition function can be split into two parts, symbolizing non-transcribing and transcribing states,

$$Z = \sum_{s} Z(s) = Z(\text{off}) + Z(\text{on}), \tag{2}$$

where $Z(\text{off})$ is the sum of all non-transcribing states and $Z(\text{on})$ the sum of all transcribing states. The probability of transcribing with a set of transcription factors able to bound to the promoter is given by,

$$P(\text{on}) = \frac{Z(\text{on})}{Z(\text{off}) + Z(\text{on})}. \tag{3}$$

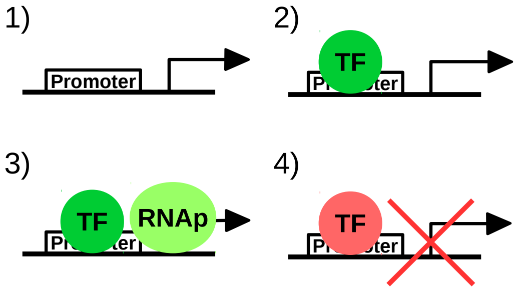


Figure 4: A schematic illustration of transcription. The gene is transcribed in state 3. Green TF characterize an activator, and red TF characterize a repressor. The arrow indicate the direction RNAp walks during transcription.

Describing the transcription process mathematically is now plausible by combining two terms. First is the transcription rate, or the production of mRNA. This expression is proportional to equation (3). Second, synthesized mRNA is not always translated to a protein. It may decay after a certain half life. This give rise to a second term, involving its degradation rate and the concentration of synthesized mRNA. Combining these two terms give the final equation, describing the dynamics of transcription as,

$$\frac{d[\Lambda]}{dt} = \alpha P(\text{on}) - \gamma[\Lambda]. \tag{4}$$

Here, $[\Lambda]$ is the mRNA concentration of gene Λ, α is the maximum rate of transcription, γ is the degradation rate, and $P(\text{on})$ is the probability of transcribing mRNA. This equation describes the dynamics of mRNA in our model. Translation of mRNA to protein is in a similar fashion defined using a partition function. The reason we exclude it here is explained in the next section.

Simplifying the rate equations

In this section, we will use equation (4) to derive a simplified rate equation of a single gene, outlining the process of modeling. The aim is to present how the partition function of Shea and Ackers was simplified. The section is concluded with a short example.

To model the gene expression of a single gene we first have to decide the scale of the model. Should it describe the process of transcription or translation, or combine the two? In this work, we model just the transcription process for two reasons. First, the protein concentrations are unknown. Consequently, describing translation would result in two unknown terms for each gene, the production and decay of its protein. Second, we wish to make the model as simple as possible. With these points in mind, the continued discussion will only consider transcription.

Describing transcription of a single gene requires knowledge of the different TFs and RNAp configurations on the promoter. Accounting for all states is impractical when constructing a simplified model. Hence, an alternative approach is to reduce the number of states by absorbing unknown terms, wherever possible, in the parameters of the model. To simplify it even further, we are not including cross terms of specific TFs in the partition function. Thus, only one activating specific TF can bind to the promoter at a time. Below, we give a brief outlining of how general TFs and RNAp are absorbed.

A state, defined by the partition function in equation (1), depends on TFs and RNAp. TFs are categorized in two groups, general and specific. Denote a general TF GTF. Assuming there is an abundance of GTFs and RNAp, in comparison to specific TFs, their concentration is constant. This means that once a specific TF is bound the assembly of GTFs and RNAp is fast. The transcription rate of a gene Λ, having factors GTF, RNAp and two specific TFs A and R, is then given as,

$$\begin{aligned}\frac{d\Lambda}{dt} =& \alpha \frac{[\text{GTF}]^{i_2}[\text{RNAp}]^{j_1}/K_2 + [\text{GTF}]^{i_3}[\text{RNAp}]^{j_2}[\text{A}]^{H_1}/K_3}{1 + [\text{GTF}]^{i_1}/K_1 + [\text{GTF}]^{i_2}[\text{RNAp}]^{j_1}/K_2 + [\text{GTF}]^{i_3}[\text{RNAp}]^{j_2}[\text{A}]^{H_1}/K_3 + [\text{R}]^{H_2}/K_4} \\ & - \gamma[\Lambda].\end{aligned} \tag{5}$$

Here, K_i represents binding affinities, A is an activator, and R is a repressor of the gene. Using the assumption for the GTF and RNAp concentrations, we can then modify the above formula by setting $[\text{GTF}]^{i_2}[\text{RNAp}]^{j_1}/K_2 = b'$, $[\text{GTF}]^{i_3}[\text{RNAp}]^{j_2}/K_3 = s_1'$, and $K_4^{-1} = s_2'$ representing background production and binding affinities. The modified rate equation looks like

$$\frac{d[\Lambda]}{dt} = \alpha \frac{b' + s_1'[\text{A}]^{H_1}}{1 + [\text{GTF}]^{i_1}/K_1 + b' + s_1'[\text{A}]^{H_1} + s_2'[\text{R}]^{H_2}} - \gamma[\Lambda]. \tag{6}$$

This equation can be further simplified, since $[\text{GTF}]^{i_1}/K_1$ is now a constant. Breaking out $1 + [\text{GTF}]^{i_1}/K_1$ plus omitting the square brackets and rewriting the modified binding affinities, yield the further simplified rate equation as,

$$\frac{d\Lambda}{dt} = \alpha \frac{b + (s_1 \cdot \text{A})^{H_1}}{1 + b + (s_1 \cdot \text{A})^{H_1} + (s_2 \cdot R)^{H_2}} - \gamma\Lambda. \tag{7}$$

This is the type of simplified rate equations we use, involving only specific transcription factors. In the case of pure repression (A= 0), the b-dependent terms can be absorbed by α. The binding affinities are again scaled. Hence, for pure repression, the resulting simplified rate equation looks like

$$\frac{d\Lambda}{dt} = \alpha' \frac{1}{1 + (s_2'' \cdot R)^{H_2}} - \gamma\Lambda. \tag{8}$$

The simplifications performed here are the notation we will use from this point forward.

As an example, consider the PTB gene previously described in section 1.1.1, seen in figure 1. It has one node affecting it, microRNA miR-124, which acts as a repressor. The transcription rate of PTB is in this case described by equation (4) with $P(\text{on}) = \frac{1}{Z}$. Using the shorthand notation of equation (8) the final expression becomes,

$$\frac{d\text{P}}{dt} = \alpha \frac{1}{1 + (s \cdot \text{miR-124})^{H}} - \gamma\text{P}. \tag{9}$$

Here, miR-124 is only found in the denominator since it represses PTB. Note that the background term has been absorbed in α, as previously mentioned. Using this type of formalism, we can calculate the time evolution for a set of rate equations that describe a regulatory gene network, see sections 2.2 and 2.3.

2.1.2 Complex formation

The REST complex is, in this model, constructed from two parts, SCP1 and REST. A method of modeling such a complex is found in Olariu et al.[42] First, denote the unbound concentration of REST and SCP1 for $[R_{\text{free}}]$ respectively $[S_{\text{free}}]$, and the bound or complex concentration for $[R|S]$. Second, assume the system is in equilibrium. The rate of bound and unbound REST and SCP1 form the following reaction equation,

$$[R_{\text{free}}] + [S_{\text{free}}] \underset{k_-}{\overset{k_+}{\rightleftharpoons}} [R|S]. \tag{10}$$

Here, k_+ and k_- represents the production rate respectively degradation rate of the complex. In equilibrium, the left hand side and right hand side are equal. The dimerization constant, K_d, is the ratio between free and bound factors,

$$K_d \equiv \frac{k_-}{k_+} = \frac{[R_{\text{free}}] \cdot [S_{\text{free}}]}{[R|S]}. \tag{11}$$

For simplicity, we assume the concentration of R and S is constant such that

$$[R_{\text{total}}]{=}[R_{\text{free}}] + [R|S], \tag{12}$$

$$[S_{\text{total}}]{=}[S_{\text{free}}] + [R|S]. \tag{13}$$

Inserting equations (12) and (13) back into (11), and solving for the REST complex yields,

$$[R|S] = \frac{K_d + [R_{\text{total}}] + [S_{\text{total}}]}{2} - \sqrt{\left(\frac{K_d + [R_{\text{total}}] + [S_{\text{total}}]}{2}\right)^2 - [R_{\text{total}}][S_{\text{total}}]}. \tag{14}$$

Note that only the negative solution is valid. For the positive solution $[R|S]$ grows with K_d so that $[R|S] \geq \frac{[R_{\text{total}}]+[S_{\text{total}}]}{2} + \frac{|[R_{\text{total}}]-[S_{\text{total}}]|}{2} = \max\{R_{\text{total}}, S_{\text{total}}\}$, but $[R|S]$ can never exceed $\min\{R_{\text{total}}, S_{\text{total}}\}$.

2.2 Parameter optimization

In this section we will present the methods used to find suitable parameters for a set of rate equations. Specifically, we will describe the methods in relation to the best performing regulatory gene network, see figure 7 and rate equations (26)-(31) on page 28. In total, this model has 45 parameters, one from the REST inhibition RESTi, one from the dimerization constant K_d of RESTc, 13 from the binding affinities s_i, 5 from the background productions b_i, 6 from the maximum transcription rates α_i, 13 from the exponents H_i, and 6 from the degradation rates γ_i. The time evolution of the genes for a specific parameter set can then be obtained through the rate equations. Finding a set of parameters that result in curves similar to experimental data constitute a challenging task. Solutions are points in the space spanned by the parameters characterizing satisfactory time evolutions of the genes. Two common strategies for finding such solutions are genetic algorithms and the simulated annealing approach. These methods use clever techniques to relatively quickly sample the high dimensional space spanned by the parameters. In this study, we use both a simulated annealing[43–46] and a genetic algorithm[47] approach to find a suitable solution.

2.2.1 Genetic algorithms

Based on natural selection, genetic algorithms simulate the process of biological evolution on a population. An initial set ("population") of candidate solutions are generated. Each solution ("individual") corresponds to a set of parameters for the system. Using a fitness function, each individual is ranked depending on their respective performance. This ranking is used to increase or decrease the chance for an individual to generate ("breed") a new individual ("child") in a stochastic selection process. Those that perform better are more likely to be selected. A child is produced by applying two operations on the selected individuals ("parents"). These two operations are crossover and mutation operations. Crossover is the process of combining the two parents, creating new children that share characteristics with the parents. Mutation symbolizes a small change of some inherited characteristic, having low probability.

In its most basic form, an individual can be illustrated as a binary string of 0:s and 1:s.[47] For example, let an individual constitute 10 parameters, each able to attain integer values up to 32. In this case, using binary representation each parameter is characterized by 5 bits ($2^5 = 32$). Hence, the individual is represented by a 50 ($5 \cdot 10$) bit string. A simple method for selecting parents to breed involve choosing the best individuals from a subset of the population. This method is known as tournament selection. Once a set of parents have been selected the two genetic operations are performed. An easy to implement crossover method would be to choose a random recombination point where the strings are to be split. This type of crossover is known as 1-point crossover. Taking one part of each parent, two new strings of 50 bits are created. Note that it is important to match the strings such that the offspring retains the string size of the parents. These constitute 10 parameters, however the children created are now each a combination of the parents. The final genetic operation involves mutating each bit by flipping the value from either 0 or 1 to the corresponding 1 or 0 with a low probability. In this way a new individual is produced in the genetic algorithm. The genetic algorithm continues to generate new populations with better fitted individuals until a set number of generations have been produced or until a stopping criterion is fulfilled. For more information of the specific individual representation, fitness function, selection process, crossover and mutation operations we used, see appendix A.

2.2.2 Simulated annealing

The main idea behind simulated annealing is to sample many candidate solutions ("states"), accepting the majority of them at the beginning then gradually decreasing the acceptance rate while favoring improving states, settling the system in a stable state. Using a parameter ("artificial temperature"), the algorithm control the acceptance rate, simulating a process of first heating the system and gradually cooling it. This action allows the system to explore several different states. Calculating the performance ("artificial energy") for each state, the algorithm can separate states from each other. Thus, gradually settling the system in a state with low energy as the temperature is cooled. By allowing energetically unfavorable moves the algorithm can avoid local minima. In this way, simulated annealing is able to find global minima.

The general scheme used in a simulated annealing algorithm consist of four steps. In the first step a random state is chosen. In our context this translates to a set of parameter values corresponding to the binding affinities (s_i), Hill coefficients (H_i), background productions (b_i), degradation rates (γ_i), and maximum production rates (α_i). To calculate the performance of the generated state, a function that can translate each state to an artificial energy is required. This function is often denoted the objective- or cost function. The cost function is problem specific, and for this reason also customizable. The notation for the cost of a state is $E(s)$, where s is the state and E the cost function. The different states form, through the cost function, an energy landscape with hills and crevasses. The temperature is an artificial parameter that controls the acceptance rate or movement in this energy landscape.

The second step of the algorithm is to generate a new candidate, also known as neighboring state s'. Commonly, this state is obtained by changing a subset of parameters using the present state. The new state is associated with a new energy $E(s')$. To traverse the energy landscape by sampling states, a function that can generate new states from the present state is required. This function is often denoted the candidate- or neighborhood function, the function that generates s'. The neighborhood function, like the cost function, is problem specific. An efficient neighborhood function can increase the algorithm performance significantly.

At the third step, the decision to either accept or reject the neighboring state is made. Based on the Metropolis algorithm, Simulated Annealing will accept or reject the neighboring state with probability $e^{-\Delta/T}$, where $\Delta = E(s') - E(s)$, and T is the given artificial temperature at step t. The calculated value is compared to an uniformly drawn random number in the interval $(0, 1)$. If $e^{-\Delta/T}$ is larger than this random number, the state is accepted. Note that state s' is always accepted if $E(s') < E(s)$, and for $E(s') > E(s)$ there exists a probability to accept state s'.

In the fourth step, the temperature is lowered. This lowers the acceptance of new states with higher energy than the current state since the decision is dependent on the temperature. Here, the system should be in thermodynamic equilibrium at all times. In practice, lowering the temperature is accomplished in discrete steps, consequently each cooling step is associated with a relaxation time. This is the time it takes for the system to reach a steady state again. For this reason, the temperature has to be lowered carefully using a cooling scheme. The cooling scheme is, like the cost function and the neighborhood function, problem specific. The goal is to have as rapid cooling as possible while avoiding local optima.

Together these four steps form the central idea of simulated annealing. A summary of them are given below.

1. Generate an initial state s, temperature T, and calculate the cost $E(s)$ associated with state s.

2. Generate a neighboring state s' and calculate its cost $E(s')$.

3. Accept or reject the neighboring state depending on the cost difference $\Delta = E(s') - E(s)$, and temperature T. The new state is accepted with probability $e^{-\Delta/T}$. If the

thermalization criterion or stop criterion is not fulfilled, go to step 2.

4. Lower the temperature T and repeat from step 2 above. If the stop criterion is fulfilled then stop.

Notice that step 2 and 3 are repeated many times before step 4. This is to thermalize the system at each temperature, allowing it to reach a steady state. This also allows the system to sample a larger region of the parameter space, which lower the probability of the algorithm getting trapped in a local optima. For more information of the specific cost function, cooling schedule and neighborhood function we used see appendix A.

In light of the algorithm descriptions given above, performing a simulated annealing or genetic algorithm parameter search on the regulatory gene network requires initial conditions, as well as unknown parameters, to be defined. We initialize each gene with the level of concentration in the corresponding data point of that gene at time $t = 0$. This method proved to give the best results in silico, but does not include the 3 days knockdown period of REST, explained in section 3.1. Another important point is that the first data point is not fitted using this method. This means that some information is lost due to excluding this data; on the other hand, this method facilitates the parameter search in the region of interest. The reason for this is that all nodes have been set to a concentration appropriate for studying the dynamics over the critical time period.

2.3 Stochastic approach

Reactions at the cellular level depend on several factors, such as the surrounding environment and the involved electrical charges of the species reacting. In addition, reactions have a probability to occur which depends on the concentration of each species in the reaction. The deterministic approach, while able to provide curves that fit data relatively well, does not capture the stochastic nature of the underlying processes. For this reason, a stochastic approach is desirable. The Gillespie algorithm provides the framework to simulate the time evolution of a chemically reacting system[48]. In this section a short derivation and description of the algorithm is given.

In order to simulate the time evolution, two questions have to be answered. Given that the system is in a state $(X_1, \ldots, X_n)$ at time t, where X_i represent concentration of species i, at what time does the next reaction occur and what type of reaction is it? To answer these two questions, Gillespie introduces the likelihood of a specific reaction μ to occur in an infinitesimal time interval dt given the system is in a state $(X_1, \ldots, X_n)$ at time t.

$$a_\mu dt = \text{Probability of reaction } \mu \text{ to occur in the infinitesimal time interval } dt\text{, given the system is in a state } (X_1, \ldots, X_n) \text{ at time } t. \tag{15}$$

Note that a_μ will be a combination of the number of available reactants, X_i, and the average probability that any distinct molecular pair react with each other. μ represents an integer number between 1 and M, where M is the number of possible reactions. The second important definition is the reaction probability density function, $P(\tau, \mu)$, defined as,

$$P(\tau, \mu)d\tau = \begin{array}{l}\text{The probability of the next reaction} \\ \text{being of type } \mu \text{ and for it to occur} \\ \text{in the infinitesimal time interval } (t+ \\ \tau, t+\tau+d\tau).\end{array} \tag{16}$$

Here, τ is a continuous variable of time, $(\tau \geq 0)$. Notice that if τ and μ are known, the answer to the two questions is known. Using definition (15), it is now possible to calculate definition (16) as the product of $a_\mu d\tau$, the probability of a reaction to occur at time τ, and the probability, $P_0(\tau)$, of no reaction to occur in time interval $(t, t+\tau)$.

$$P(\tau, \mu)d\tau = P_0(\tau) \cdot a_\mu d\tau. \tag{17}$$

The expression of $P_0(\tau)$ is deduced by realizing that for no reaction to occur in a time $d\tau$ the probability is $[1 - \sum_\nu a_\nu d\tau]$, where ν is an index from 1 to M. The probability that no reaction occurs in a time interval $(\tau, \tau + d\tau)$ is,

$$P_0(\tau + d\tau) = P_0(\tau)\left[1 - \sum_{\nu=1}^{\mathrm{M}} a_\nu d\tau\right]. \tag{18}$$

The expression of $P_0(\tau)$ is derived by dividing both sides in (18) with $d\tau$, and rearranging the equation, such that,

$$\frac{P_0(\tau + d\tau) - P_0(\tau)}{d\tau} = -P_0(\tau)\sum_{\nu=1}^{\mathrm{M}} a_\nu. \tag{19}$$

The above equation defines the derivative of $P_0(t')$ with respect to time, meaning,

$$\frac{dP_0(\tau)}{d\tau} = -P_0(\tau)\sum_{\nu=1}^{\mathrm{M}} a_\nu. \tag{20}$$

Solving (20) yield the expression,

$$P_0(\tau) = e^{-\sum_{\nu=1}^{\mathrm{M}} a_\nu \tau}. \tag{21}$$

Inserting expression (21) into equation (17) and integrating both sides yield the final expression of the probability $P(\tau, \mu)$,

$$P(\tau, \mu) = \begin{cases} a_\mu e^{-\sum_{\nu=1}^{\mathrm{M}} a_\nu \tau}, & \text{if } \tau \geq 0 \text{ and } \mu = \\ & 1, \ldots, M, \\ 0, & \text{otherwise.} \end{cases} \tag{22}$$

Note that $P(\tau, \mu)$ depends on all reactants at every time t. From here on, $a_0 = \sum_{\nu=1}^{M} a_\nu$, will be used. Determining the two quantities τ and μ answer the questions stated previously:

at what time does the next reaction occur and what type of reaction is it. Generating τ and μ such that they follow the probability distribution (22) ensures that reactions occur statistically and in proportion to their concentration. The two variables are generated by the two equations below,

$$\tau = \frac{1}{a_0} \ln \left(\frac{1}{r_1} \right), \tag{23}$$

$$\sum_{\nu=1}^{\mu-1} a_\nu < r_2 a_0 \leq \sum_{\nu=1}^{\mu} a_\nu. \tag{24}$$

Here, $r_{1,2}$ are uniformly drawn pseudorandom numbers in the interval $(0, 1)$. In equation (24), μ is the first index of summing a_ν such that their sum exceeds the product between the pseudorandom number r_2 and a_0. The derivation of these two equations can be found in Gillespie[49].

The two questions stated previously can now be answered. Using the two formulas (23) and (24), we know at what time the next reaction will occur and what type of reaction it will be. The steps to simulating the regulatory gene network stochastically can now be summarized in a pseudocode as the following,

1. Initialize time $t = t_0$, and the concentrations $(X_1, \ldots, X_n)$.
2. Calculate the total probability density sum a_0 of all reactions.
3. Derive τ and μ using equations (23) and (24).
4. Update the system, $t = t + \tau$, and $X_i = X_i + x_\mu$, where X_i is the current concentration of gene i and x_μ is a concentration change for that gene.
5. Repeat from step 2 or exit.

For the regulatory gene network, the mRNA of a gene is involved in two types of reactions, production or degradation. In the Gillespie algorithm, production yields a positive contribution, $x_\mu > 0$, and degradation a negative contribution, $x_\mu < 0$. In total, the number of possible reactions, M, are twice as many as the number of rate equations used.

2.4 Reduced stochastic noise

Working with the stochastic implementation, the standard concentration change of unity, often chosen for the Gillespie algorithm, proved to yield too extreme fluctuations. The reason for this is that in comparison to the concentration levels measured experimentally, a change of unity is very large. Thus, each concentration increment or decrement in the stochastic simulation has to be scaled such that the noise does not overpower the underlying dynamics. In order to reduce the noise, we modify both the level of expressed mRNA and time increment by a constant c. This preserves the scaling of the system to fit with the normalized data. Denote the current level of mRNA of a gene for X_i, then the modified mRNA expression and time increment used is given by,

$$\begin{cases} X_i = X_i + c \cdot x_\mu, \\ \tau = \dfrac{c}{a_0} \ln\left(\dfrac{1}{r_1}\right), \end{cases} \quad 0 \leq c \leq 1. \tag{25}$$

Here, x_μ is the standard unity change and τ the time increment, both multiplied by the same constant c. This is to ensure that on average the same amount of reactions occur in the time period. Determining the constant c is done by trial and error. In this work $c = 0.01$.

3 Results

3.1 Experimental data

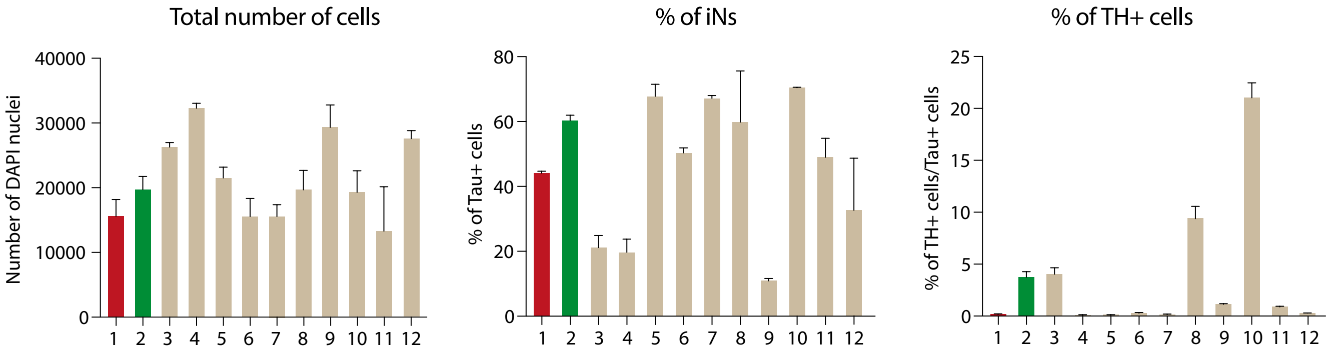
Experiments injecting genetic material in the cell have been conducted since the mid-to-late 20th century[50,51]. In the case of cell conversion, these experiments may test different sets of reprogramming factors trying to find possible improvements in the differentiation scheme and quality of the differentiated cells[52]. In this section, the experimental data is explained in more detail.

Our collaborators Malin Parmar and Janelle Drouin-Ouellet have performed an in vitro study using different viral vectors on human adult fibroblast cells[25]. The experiments were conducted using HAF cell culture in petri dishes that are injected with a reprogramming cocktail. By mounting the vectors with different reprogramming factors, different neuronal conversion schemes can be studied. The mRNA concentration of a variety of genes is then monitored during a 21 day period. The group used a total of 12 different cocktails with varying degree of success. Results show that using a cocktail consisting of Ascl1, Lmxa1, Lmxb1, FoxA1, otx2, Nurr1, and RESTi, performed the best. Here, Lmxa1, Lmxb1, FoxA1, otx2, and Nurr1 are factors that facilitate production of dopamine of the converted neurons, these are not included in our model.

The data of the 12 different cocktails are shown in figure 5. Here, the first histogram to the left illustrates approximately the number of cells in use for each cocktail. The histogram in the middle shows the percentage of cells that transdifferentiate to a neuron-like state, in relation to the initial number of cells. Note that cocktail number 10 performed the best with a roughly 70% conversion rate. The last histogram indicate the percentage of the transdifferentiated neuron-like cells that produce TH, an enzyme that facilitates production of dopamine[53].

Cells classified as not converted constitute both HAF cells and other types of immature neurons which did not transdifferentiate accordingly. The reprogramming results in many different types of neuron topologies. In this study, we do not make a distinction between different neurons. The only criterion is that the neurons display normal physiology. In the experiment, cells were classified as neuronal and non-neuronal by marking a protein characteristic for neurons with green fluorescence protein (GFP) and sorting cells expressing this protein. Experimental data shows cocktail number 10 as being the best candidate to achieve high conversion. Therefore, we use data gathered from cells grown on medium with this cocktail to model transdifferentiation.

In figure 6 the time series of the mRNA concentration for the genes used in our model are shown. Here, the orange dots connected with orange lines represent the median value. Every time t has at most three measurements associated with it on the vertical axis. The pale yellow shaded area illustrates the spread of data, meaning the highest respectively lowest measured concentration at every median value used. The time series shown was constructed by calculating an average between the present number of neurons and the



1. Ascl1_Brn2_RESTi (negative control)
2. Ascl1 + Lmx1a + Lmx1b + FoxA2 + otx2 + RESTi (positive control)
3. Ascl1_Brn2_RESTi + Lmx1a + Lmx1b + FoxA2 + otx2
4. Ascl1_Lmx1a
5. Ascl1_Lmx1a + RESTi
6. Ascl1_Lmx1a + Ascl1_Lmx1b + RESTi
7. Ascl1_Lmx1a + Ascl1_FoxA2 + RESTi
8. Ascl1 + Lmx1a + Lmx1b + FoxA2 + otx2 + Smarca1 + RESTi
9. Ascl1_Brn2_RESTi + Lmx1a + Lmx1b + FoxA2 + otx2 + Smarca1 + RESTi
10. Ascl1 + Lmx1a + Lmx1b + FoxA2 + otx2 + Nurr1+ RESTi
11. Ascl1_Lmx1a + Ascl1_Lmx1b + Ascl1_FoxA2 + Otx2 + RESTi
12. Ascl1 + Lmx1a + Lmx1b + FoxA2 + CNPY1+ En1+ Pax8 + RESTi

Figure 5: Effectiveness of 12 different viral vectors tested experimentally[25]. First histogram (left) displays initial number of cells. Second histogram (middle) displays the neuron conversion rate of each vector from the first histogram. Third histogram (right) displays the percent of cells from the second histogram that produce TH protein, a protein used in the production of dopamine. We model conversion after time series data obtained using cocktail number 10 since it performed the best.

initial number of undifferentiated cells at each time step. Each data point symbolizes a bulk concentration from an independent time series of the single petri dish. The time span of the experiment was 21 days in total, plus an initial 3 days of REST knockdown, see appendix B. The REST knockdown phase is a required step in order to suppress the repression of neural genes.

During the transdifferentiation process the concentration of each reprogramming factor in a cell is difficult to control. Studying the data dynamics leads us to believe that the important switch decisions are made within the first 5 days. Here, the time series of PTB, REST, and SCP1 display a switch behavior around day 1. First, an initial downregulation occurs between hour 0 and hour 8, followed by an upregulation till roughly day 1 and a downregulation from that point to a steady state by day 5. Viewing the data, it appears that this switch is an important step for the cell to convert. In a similar fashion, a switch occurs around day 1 for neuronal genes nPTB, miR-124, and miR-9. After 5 days the system seems to settle into a steady state with Ascl1, nPTB, and miRs high, see appendix B. In the 5 days time period a trend is clearly visible; neuronal genes, endogenous Ascl1, nPTB, miR-124, and miR-9, are being upregulated while the non-neuronal genes, PTB, SCP1, and REST, are being downregulated. However there exists some flexibility in the

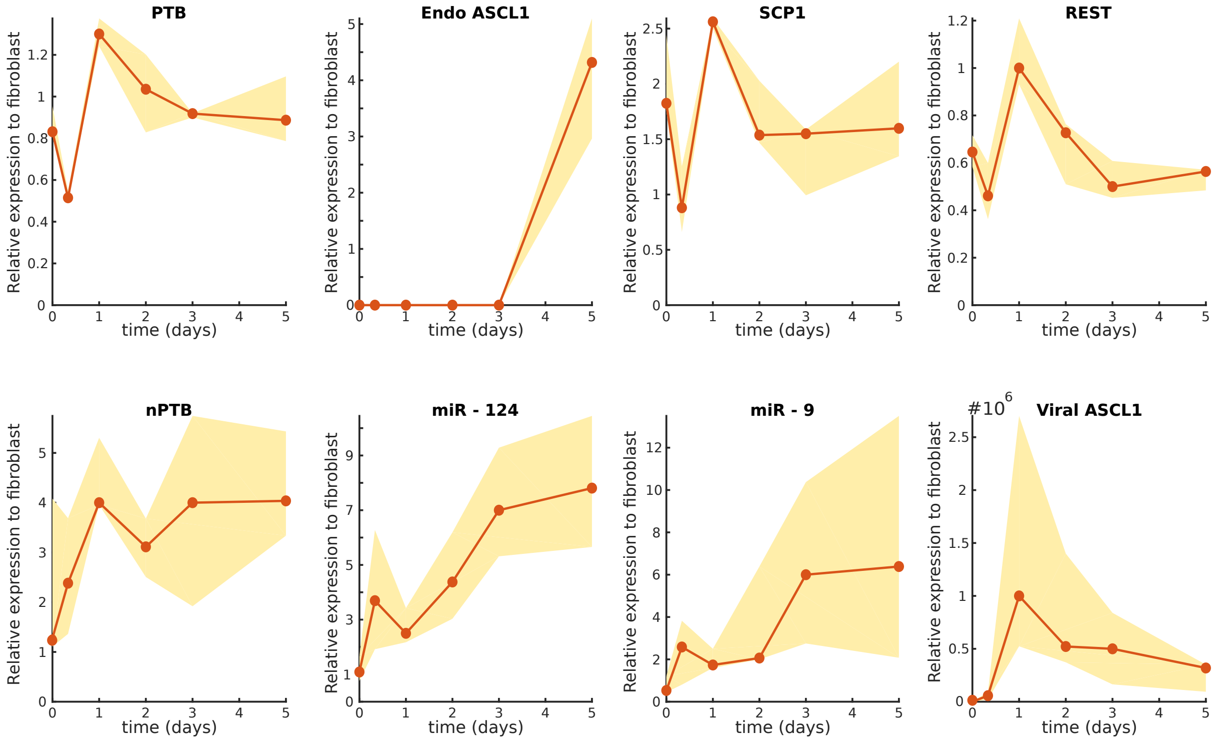


Figure 6: Time series data showing the gene expression level for the genes in the model. The pale yellow shaded area illustrates the maximum respectively the minimum values measured during the experiment. The orange dots connected with orange lines illustrate the median value at each measurement. Measurements shown exist at times $(0, 8, 24, 48, 72, 120)$ measured in hours, converted to days on the x-axis. Every time t has at most three measurements associated with it on the vertical axis. Note that this data is normalized such that a value of 1 signifies the normal concentration in a fibroblast cell. Data shown here is for the first 5 days, preceded by 3 days of REST knockdown.

exact dynamics since the time series are composed of independent time series. For this reason, the true shape of the transdifferentiation curves is unknown. From the middle histograms in figure 5, we know that most cells transdifferentiate. Hence, we denote the time series neuronal states. We fit the model to the median value, the orange data points, at times $(0, 8, 24, 48, 72, 120)$ in hours. Time evolutions produced by the model deviating from these states (meaning low concentration of Ascl1, miR-124, miR-9, nPTB, and high concentration of REST, SCP1, and PTB), we denote non-neuronal states. Distinguishing between the two states will be discussed further in section 3.4.

In summary, the best performing cocktail consisted of REST inhibition and Ascl1 overexpression. These factors are transducted in the cell via viral vectors. The conversion rate to neuronal-like cells using this cocktail was roughly 70%. REST inhibition and viral Ascl1 are denoted RESTi and virA in our model. The experimental time series data are composed from many independent time series. Each data point symbolizes a bulk concentration of the cells in the petri dish. The critical transdifferentiation processes occur in the first 5 days period.

3.2 The regulatory gene network

This section will aim to further explain how we model the best performing regulatory gene network, seen in figure 7 with equations (26)-(31). Here, we provide the motive for introducing hypothetical interactions, the classification of neuronal and non-neuronal states, treating viral and endogenous Ascl1 separately, merging the microRNA nodes, and what the model symbolizes. The section is concluded with a list of the main assumptions of the model.

Merging the two regulatory loops from section 1.1.1 with the interactions described in 1.1.2, resulted in a network which gave unsatisfactory fits to the time series. In order to improve the performance of the regulatory gene network, we modified its topology. Previously, our collaborators found RESTc to be a key barrier in the reprogramming[25]. For this reason, we focus on modeling interactions to and from RESTc and its components in silico. By trial and error, we tested different hypothetical interactions in order to find a set that yields a relatively good fit to data, see appendix C. We tried various combinations systematically, focusing on RESTc and its components. Fitting both SCP1 and REST to data proved challenging as their central role in the network affects the fitting of all other genes. RESTc interacting with its components improved the ability for the regulatory gene network to fit data. Combining complex-to-component interactions with other interactions we were able to find a winning network.

The best performing topology is determined by having the lowest cost or fitness, and by not having an apparent unphysical behaviour. The winning topology we found is shown in figure 7. The hypothetical interactions involve activation of PTB, SCP1, and REST from RESTc, as well as activation of SCP1 and Ascl1 from nPTB. Note that all of these interactions (except the Ascl1 activation) act to facilitate the plasticity of the involved genes, PTB, SCP1, and REST. In the figure, rectangular nodes represent the experimental cocktail used for transdifferentiation. REST inhibition is denoted short hairpin REST, shREST, known as RESTi in the rate equations.

The set of rate equations of the winning topology is given by equations (26)-(31). Here, P, A, nP, and virA, denote PTB, endogenous Ascl1, nPTB, and viral Ascl1. Viral Ascl1 as well as RESTi are transducted into the cell via viral vectors, explained in section 3.1. The two microRNAs 124 and 9 have been merged to a single node, denoted miR. The reason for this merge is explained in the next paragraph. In the rate equations, γ_i represents inverse half-life times, s_i represents binding affinities, b_i represents background productions, α_i represents maximum transcription rates, RESTi represents REST inhibition, and H_i represents how many factors bind to the promoter. From a modeling perspective this term represents the degree of non-linearity. In total, the model has 45 parameters.

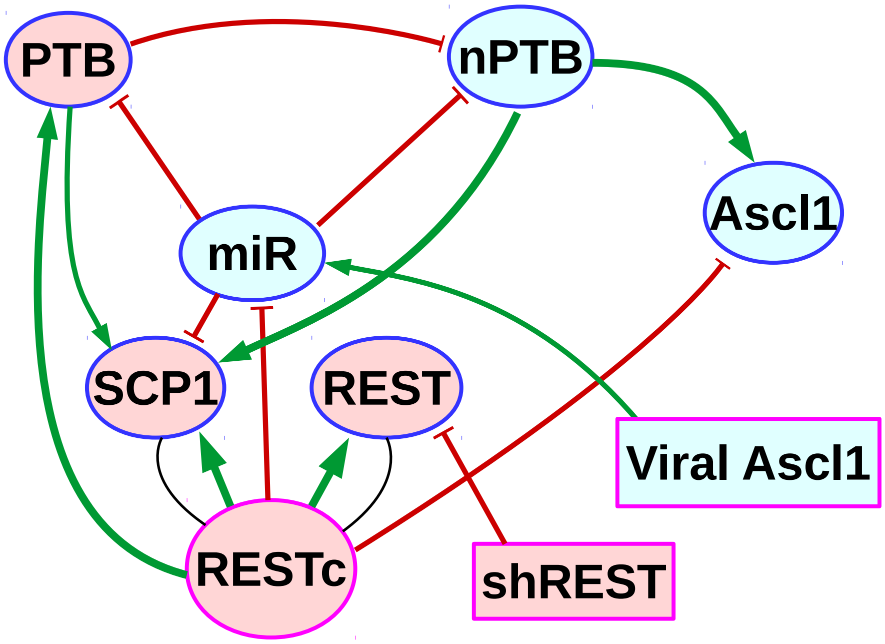


Figure 7: Topology of the best performing regulatory gene network. Here, green connecting arrows symbolize activation and red connecting arrows symbolize repression. Note the three thicker activations from RESTc and the activation from nPTB to SCP1 and Ascl1. These are hypothetical interactions added to the network. REST inhibition is denoted short hairpin REST, shREST. The red transparent nodes portrays non-neuronal genes and cyan transparent nodes portrays neuronal genes. Nodes with blue contour have a rate equation and those with purple contour have not. Oval nodes characterize endogenous genes, expressed by the cell, and rectangular nodes are transducted factors, inserted externally in the cell.

$$\frac{d\mathrm{P}}{dt} = \alpha_1 \frac{b_1 + (s_1 \cdot \mathrm{RESTc})^{H_1}}{1 + b_1 + (s_1 \cdot \mathrm{RESTc})^{H_1} + (s_2 \cdot \mathrm{miR})^{H_2}} - \gamma_1 \cdot \mathrm{P} \tag{26}$$

$$\frac{d\mathrm{A}}{dt} = \alpha_2 \frac{b_2 + (s_3 \cdot \mathrm{nP})^{H_3}}{1 + b_2 + (s_3 \cdot \mathrm{nP})^{H_3} + (s_4 \cdot \mathrm{RESTc})^{H_4}} - \gamma_2 \cdot \mathrm{A} \tag{27}$$

$$\begin{aligned}\frac{d\mathrm{SCP1}}{dt} &= \alpha_3 \frac{b_3 + (s_5 \cdot \mathrm{P})^{H_5} + (s_6 \cdot \mathrm{RESTc})^{H_6} + (s_7 \cdot \mathrm{nP})^{H_7}}{1 + b_3 + (s_5 \cdot \mathrm{P})^{H_5} + (s_6 \cdot \mathrm{RESTc})^{H_6} + (s_7 \cdot \mathrm{nP})^{H_7} + (s_8 \cdot \mathrm{miR})^{H_8}} \\ &\quad - \gamma_3 \cdot \mathrm{SCP1}\end{aligned} \tag{28}$$

$$\frac{d\mathrm{REST}}{dt} = \alpha_4 \frac{b_4 + (s_9 \cdot \mathrm{RESTc})^{H_9}}{1 + b_4 + (s_9 \cdot \mathrm{RESTc})^{H_9} + \mathrm{RESTi}} - \gamma_4 \cdot \mathrm{REST} \tag{29}$$

$$\frac{d\mathrm{nP}}{dt} = \alpha_5 \frac{1}{1 + (s_{10} \cdot \mathrm{miR})^{H_{10}} + (s_{11} \cdot \mathrm{P})^{H_{11}}} - \gamma_5 \cdot \mathrm{nP} \tag{30}$$

$$\frac{d\mathrm{miR}}{dt} = \alpha_6 \frac{b_5 + (s_{12} \cdot \mathrm{virA})^{H_{12}}}{1 + b_5 + (s_{12} \cdot \mathrm{virA})^{H_{12}} + (s_{13} \cdot \mathrm{RESTc})^{H_{13}}} - \gamma_6 \cdot \mathrm{miR} \tag{31}$$

The two microRNAs, miR-124 and miR-9, described in earlier sections, have interactions of similar nature. Both microRNAs regulate similar genes. For example, miR-124 and miR-9 have targeting sites on nPTB[24], as mentioned in section 1.1.1. In addition, miR-124, and miR-9 are also repressed by the REST complex, as well as activated by Ascl1. Observing the time series of the microRNAs in figure 6, the curves have almost identical dynamics. For these reasons, we combine them into a single node that is denoted miR. This reduces the number of parameters in the network, facilitating the process of fitting network topologies to data. Merging miR-124 and miR-9 to a single node constrains us to optimize this node using only one time series. The dynamics are similar between the two curves, making the choice arbitrary. Here, we choose the miR-124 data. Note that this merge is made solely for facilitating the parameter optimization.

The reprogramming cocktail we model uses transduction of Ascl1 and REST inhibition, the best performing cocktail in vitro mentioned in section 3.1. In this thesis, transducted Ascl1 is denoted viral Ascl1 (virA), and REST inhibition is denoted RESTi. RESTi is modeled as an unknown parameter in the model that appears in the denominator of equation (29). Note that short hairpin REST (shREST) in figure 7 symbolizes RESTi. This parameter represents REST knockdown. Observing equation (29), REST would evidently decrease by letting RESTi increase, resulting in low REST, as well as RESTc. Ascl1 is modeled in a different way since both viral and endogenous levels are measured in the experiment. The extreme difference in concentration levels between viral and endogenous Ascl1 are challenging to fit with a single rate equation. The reason being that fitting to viral Ascl1 immediately saturates this node, which then affects all other nodes in the network, either repressing or activating them strongly. Ultimately, it is the endogenous curve that is interesting to our model. Therefore, we separate viral and endogenous Ascl1. In the cell viral Ascl1 will still take part in the majority of interactions because of the high concentration. To account for this fact, we replace the endogenous Ascl1 term with a viral Ascl1 term in interactions that would normally, without transduction of Ascl1, occur in proportion to endogenous levels. The viral part is modeled as a constant parameter that symbolizes a saturated value of Ascl1 in the cell. We chose the value of viral Ascl1 to be 1. The endogenous part can then be represented by rate equation (27).

In contrast to the experimental time series which symbolizes bulk concentrations, the model portrays the behavior of a single cell. Therefore, we assume that the single cell gene expression resembles that of the bulk concentration. Furthermore, RESTc is modeled as composed of two components, REST and SCP1, while in nature the complex consist of more components[19].

In summary, the regulatory gene network presented here is based on key assumptions derived from the methods we use and experimental data. These assumptions are listed below.

- Viral and endogenous Ascl1 are separate and viral Ascl1 is constant throughout experiment.
- Bulk time series are indicative of single cell time series.

- The essential steps in transdifferentiation occur in the first 5 days.
- microRNA 124 and 9 have similar dynamics and can be merged to a single node.
- RESTc is composed of two components, REST and SCP1.

3.3 Fitting the network to time series

Determining a set of parameters for a given topology is achieved by fitting equations (26)-(31) with the corresponding time series in figure 6. We use simulated annealing and genetic algorithm for the parameter optimization. The genetic algorithm performed the best, results are shown in figure 8. Orange dots characterize median values from figure 6. Blue curves characterize the best deterministic fit. The corresponding set of model parameters producing these curves can be seen in appendix D.

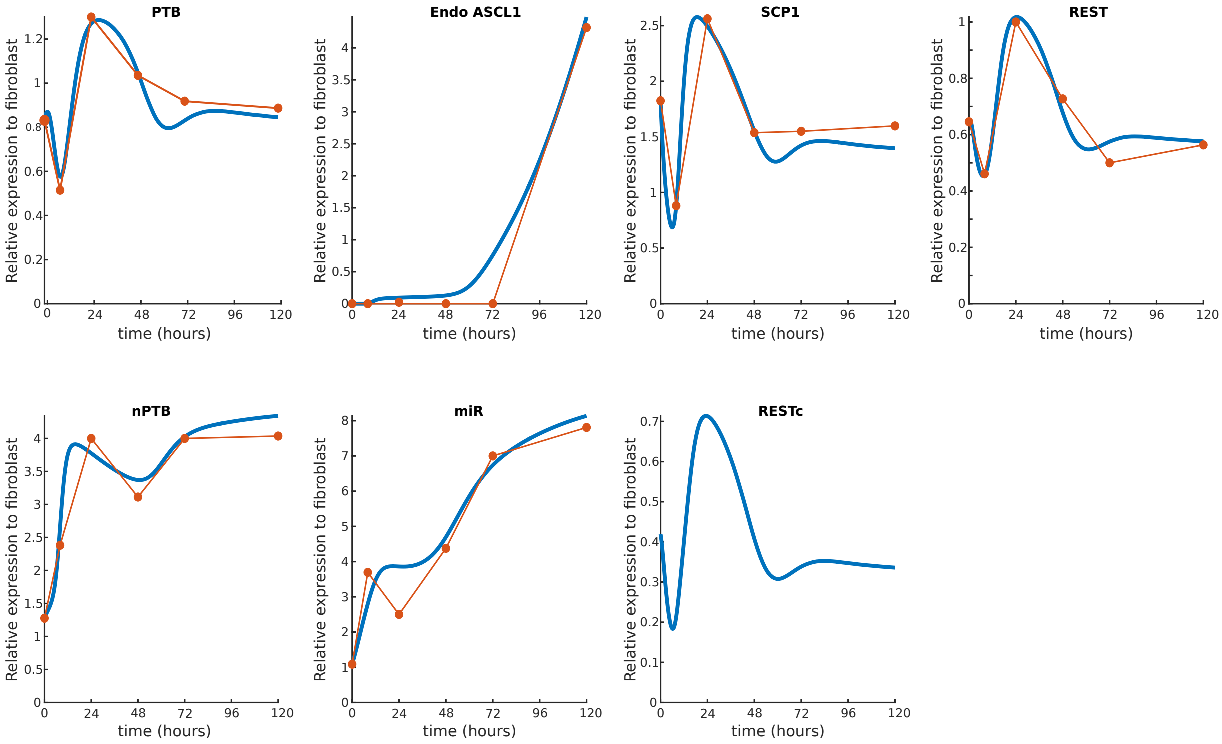


Figure 8: Best deterministic fit of winning topology. Here, orange dots depict data, connected with orange lines. Blue curves represent model simulation results with the best parameter values. The REST complex (RESTc) is included to show how it evolves as a function of SCP1 and REST.

Figure 8 shows SCP1 and REST having similar dynamics to the experimental time series. These are key regulatory genes that are important to achieve an accurate description of the underlying processes. Transduction of viral Ascl1 results initially in a sharp decrease of SCP1 and PTB through miR, seen between data points at $t = 0$ hours and $t = 8$ hours. Repressing PTB relieves the repression on nPTB, resulting in an upregulation of SCP1. A positive feedback between the complex and its components results in an upregulation of SCP1, REST, RESTc, PTB, and a downregulation of nPTB and miR. Endogenous Ascl1 level was largely unchanged, the reasons is that RESTc is still repressing it and nPTB is not activating it strongly enough. At a later stage, upregulation of miR through viral Ascl1 manages to overcome RESTc, and the non-neuronal genes are again downregulated. At this point miR has reached such a high concentration that RESTc is staying low. Endogenous Ascl1 is upregulated as nPTB keeps increasing, and RESTc keeps decreasing, seen from the data point at $t = 48$ hours and onward.

The general trend of each node has been reproduced by the corresponding deterministic solution. Possible improvements involve a better downregulation of miR at $t = 24$ hours, as well as a smoother behavior of PTB, SCP1, and REST between $t = 48$ hours and $t = 72$ hours. In the $48 - 72$ hour time region, a dimple appears, even though intuitively a smoother behavior would be more likely in this region. The experimental time series seem to suggest that the transdifferentiation process has key regulatory switches during the conversion.

The regulatory gene network represents a kind of artificial cell following the experimental bulk concentration. Figure 9 shows deterministic time evolutions with different levels of perturbation on the binding affinity of PTB on SCP1, see s_5 in equation (29). Depending on the value of the binding affinity, the artificial cell evolves into two distinct states (blue or red). In the blue state neuronal genes nPTB, miR, and endogenous Ascl1 are enhanced, while in the red state they are repressed. Non-neuronal genes PTB, SCP1, and REST showed anti-correlated behavior with the neuronal genes in each state. This rapid transition suggests that controlling SCP1 is an important part for the transdifferentiation procedure. One reason for this, is that SCP1, together with REST, builds the complex repressing the neuronal genes. Thus, it is not unlikely that to achieve high yield conversion, other complex compounds besides REST ought to be controlled. SCP1 also has a relatively central role in our model with many genes affecting it, suggesting it has a key role in the transdifferentiation.

Examination of the other binding affinities on SCP1 revealed that the RESTc binding affinity also produce a rapid transition. Increasing the effect of nPTB on SCP1 through its binding affinity yields no rapid transition. In this case nPTB induces the RESTc-PTB-nPTB pathway, repressing itself through non-neuronal genes. The miR-124 binding affinity was also unable to produce a rapid transition.

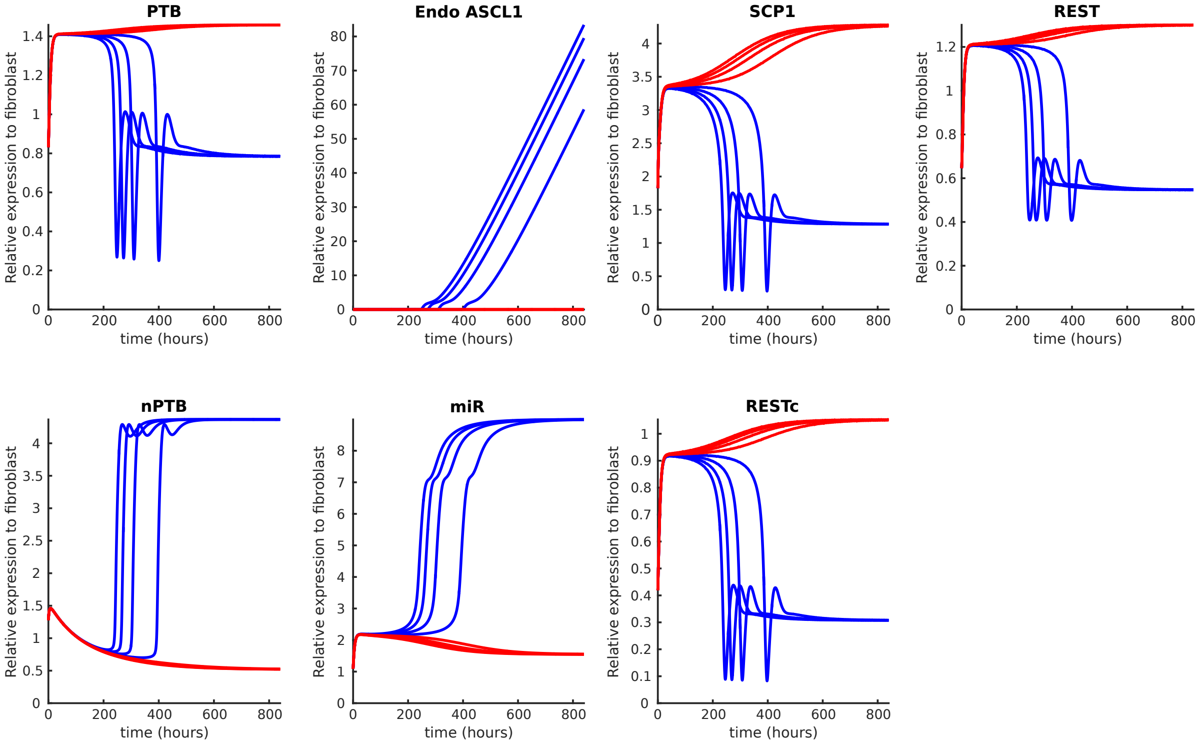


Figure 9: Deterministic time evolutions of slightly perturbing the binding affinity of PTB on SCP1, see parameter s_5 in equation (29). All other parameters were kept constant. Blue curves evolve to a neuronal like state and red curves evolve to a non-neuronal like state. Note the rapid transition seen in the system. This suggests that the network has a bistable region and that SCP1 is an important node to control during the conversion. The y-axis illustrates the relative fold to fibroblast levels and the x-axis the simulation time. The time scale has now changed as perturbing the binding affinity results in skewing the effect of PTB on SCP1. The parameter range is $s_5 \in [3.3, 3.317]$, using a step size of 0.001.

3.4 Stochastic simulations

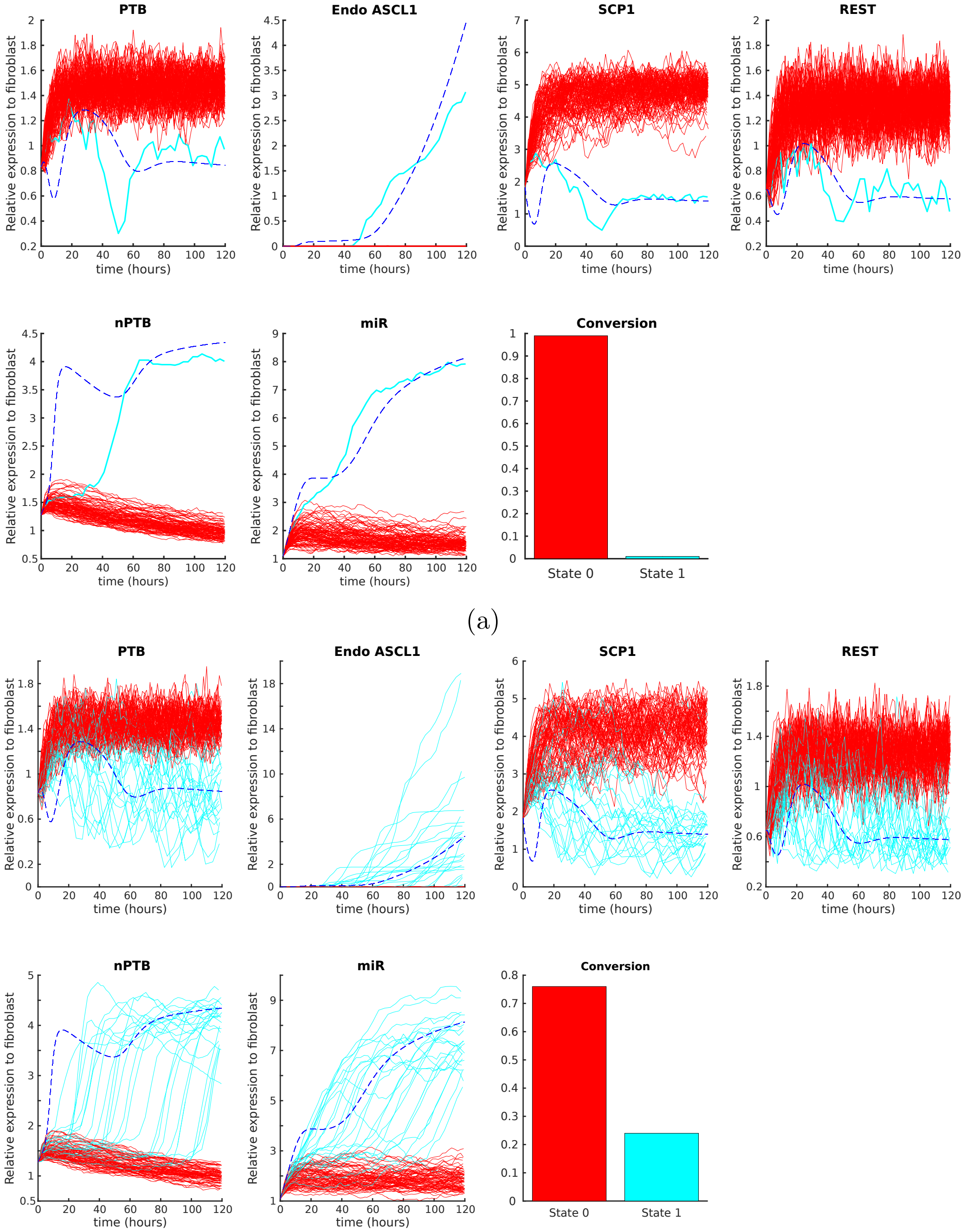
The stochastic nature of reactions result in varying time evolutions of a compound. Simulating an ensemble of cells using a deterministic approach yields no additional information, as with no noise, each cell produces the same time evolution. On the other hand, introducing noise, such that each cell in the ensemble behaves differently, yields a more accurate picture of how the regulatory gene network would behave in nature. We conducted stochastic simulations using the Gillespie algorithm, previously described in section 2.3. The simulations use the parameters of the best deterministic fit. Perturbing the binding affinity, cells differentiated into two different states, a neuronal- and non-neuronal like state. Specifically, we studied the interactions of SCP1, to gain further understanding of the importance of controlling the REST complex during transdifferentiation.

Figure 10 shows 4 different stochastic simulations, each containing an ensemble of 100 cells. All cells are initialized with the experimental bulk concentration at time $t = 0$ hours. The time span of the simulations is between $t = 0$ hours and $t = 120$ hours, the same as the deterministic simulation. In figures 10a-d, the dotted blue lines represent the deterministic fits of figure 8. These dotted curves indicate how closely a cell follows the deterministic dynamics. Figure 10b and c show cyan curves representing cells that follow the deterministic solution relatively closely, and red curves representing cells deviating comparatively strongly from this state. Cyan cells have low neuronal repression, meaning nPTB, Ascl1, and miR are upregulated, while, red cells has strong neuronal repression with REST, SCP1, and PTB upregulated. This is similar to the two deterministic states mentioned previously.

We use endogenous Ascl1 to characterize a cell as having either a high or low neuronal repression. A mean value of the concentration from the last hour is calculated for the deterministic and stochastic endogenous Ascl1. The ratio between the stochastic and deterministic mean value is compared to a threshold value. If the ratio is larger than or equal to this value, the stochastically evolving cell is counted as a neuronal cell. The threshold value is 0.01, determined by hand.

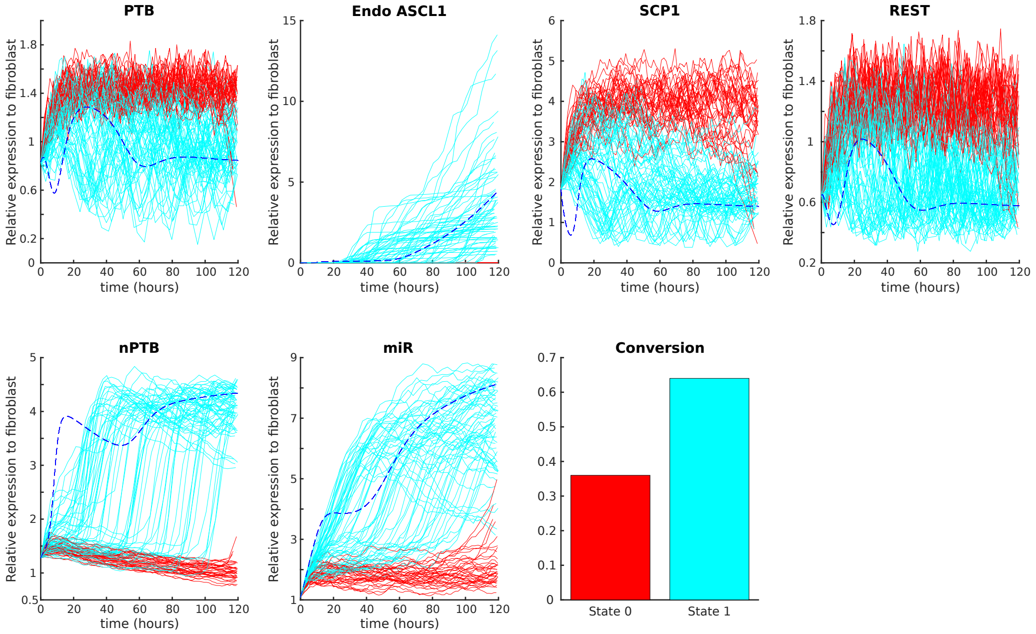
The cells in the four stochastic simulations have different binding affinities of PTB on SCP1. The values used are near the critical range found in the deterministic simulation. Figure 10a shows the result of increasing the binding affinity to a relatively high value. Increasing the binding affinity symbolizes in this case increasing the effect of PTB on SCP1. From an ensemble of 100 cells, only a single cell ends up in the neuronal-like state. Note that each figure, 10a, b, c, and d, is accompanied by a histogram, detailing the percent of cells in the respective state. The histograms use a shorthand notation of state 0 (red) for characterizing high neuronal repression and state 1 (cyan) for characterizing low neuronal repression. Decreasing the binding affinity, meaning we lower the strength of the effect on SCP1 from PTB, results in figure 10b. Here, 75 cells end in state 0 while 25 cells end in state 1. The non-neuronal state is still dominating, though lowering the binding affinity clearly produced an increase in neuronal-like cells. In figure 10c, decreasing the binding affinity even more leads to a switch with the neuronal state as the dominant attractor of the system. The conversion rate is not ideal, 65 cells end in state 1 and 35 cells in state 0. Figure 10d illustrates the time evolution of the ensemble of cells using the default binding affinity, found through the deterministic procedure. Here, 100 cells end in the neuronal state. Other sources of noise effecting the conversion, besides the background noise of Gillespie, are not included in our simulations. Including these will likely result in a lower conversion rate, closer to that observed experimentally.

In summary, transdifferentiation is in our model conducted through overexpressing Ascl1 transcription factor and repressing the gene expression of REST, based on experimental results by our collaborators[25]. Using established connections, as well as introducing hypothetical interactions for SCP1, REST, PTB and nPTB, we found a regulatory gene network able to fit experimental time series data, see figure 8 and equations (26)-(31). Further investigating the network dynamics showed that perturbing the binding affinity (s_5 in equation (29)) on SCP1 from PTB yield a rapid transition, see figure 9. This suggests that there exist at least two attractors for the regulatory gene network. Performing stochastic simulations showed that the regulatory gene network model exhibits two attractors, a neuronal- and non-neuronal. Systematically lowering the binding affinity from a relatively high value to the value found through the parameter optimization, we observed both attractors. At high respectively low binding affinity, the cells were more prone to evolve into the corresponding non-neuronal respectively neuronal attractor. Between these extremes, cells would evolve into both attractors.

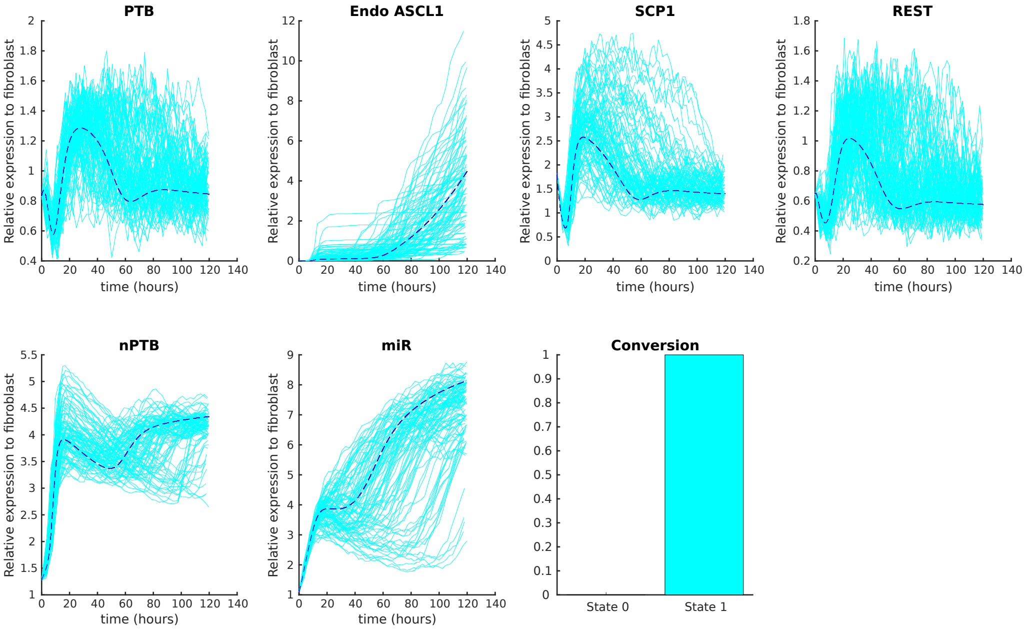


(a)

(b)


PTB
Endo ASCL1
SCP1
REST
nPTB
miR
Conversion
Relative expression to fibroblast
time (hours)
0
20
40
60
80
100
120
0.2
0.6
1
1.4
1.8
5
10
15
2
3
4
5
6
0.5
7
9
0.1
0.2
0.3
0.4
0.5
0.6
0.7
State 0
State 1

(c)


PTB
Endo ASCL1
SCP1
REST
nPTB
miR
Conversion
Relative expression to fibroblast
time (hours)
0
20
40
60
80
100
120
140
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
4
6
8
10
12
0.5
1.5
2.5
3.5
4.5
5
5.5
0.2
3
7
9
0.1
0.3
0.5
0.7
0.9
State 0
State 1

(d)

Figure 10: Stochastic simulations of 100 cells initialized with the experimental concentration at time $t = 0$. Here, the binding affinity parameter of PTB in rate equation (29), is changed between each figure. Two states can be distinguished, a neuronal- (cyan) respectively non-neuronal (red) state. Red curves represent cells in state 0, a state with high repression of neuronal genes. Cyan curves represent cells with low repression of neuronal genes, meaning they follow the deterministic solution to a degree. Histograms in each figure show the percent of cells in the respective state. For comparison, the best deterministic fit is included, blue dotted lines. (a) Highly perturbed binding affinity of PTB in the SCP1 rate equation. Here, the non-neuronal state is dominating strongly. (b) Lowering the perturbation to some degree results in 75 non-neuronal cells and 25 neuronal cells. The dominant state is still the non-neuronal state. (c) Further lowering the perturbation shifts the system. Here, neuronal state 1 is dominating, with 65 cells in this state and 35 cells in the non-neuronal state. (d) Using parameters from the best deterministic fit, all cells stay in the neuronal state. Endogenous Ascl1 is used to identify which state a cell is in. The curves are plotted from time $t = 0$ to $t = 120$.

4 Discussion and conclusions

In this thesis, we have studied direct lineage reprogramming from human adult fibroblast cells to neurons in silico. A computational model capturing the main parts of this transdifferentiation could facilitate the experimental search of a high yield conversion scheme. The advantage is that testing the dynamics of the transdifferentiation is relatively time efficient. The model is constructed on experimental results of two regulatory loops[23,24]. Merging the two loops and incorporating hypothetical interactions enabled us to find a regulatory gene network able to reproduce experimental time series. Experimental research by our colleges, Drouin-Ouellet et al.[25], suggests RESTc as a key reprogramming barrier in the transdifferentiation. Therefore, we focus on RESTc and its components when incorporating hypothetical interactions with the rest of the nodes in the network. Transdifferentiation is in our model performed by overexpressing Ascl1 and repressing REST, the best performing cocktail experimentally[25].

Using two global search methods, simulated annealing and genetic algorithm, we searched for a winning network. The winning topology is shown in figure 7, with rate equations (26)-(31). Here, the connections between RESTc and SCP1, RESTc and REST, RESTc and PTB, nPTB and Ascl1, and nPTB and SCP1, are predictions of our model. The winning set of parameters produced curves that fit well to experimental data. Some characteristics still need to be addressed: first the downregulation of miR at time $t = 24$ hours is not captured by the model. Here, the miR expression level is plateauing instead of being downregulated. A likely explanation of the plateau effect on miR is the constant activation by viral Ascl1. Second, the small dimples seen in the non-neuronal genes, PTB, SCP1, and REST, are interesting. This is likely a compensation made by the algorithm to improve the fitting, though it is possible a natural phenomenon exists behind it.

Using the rate equations and determined parameters, we performed deterministic and stochastic simulations of the system. Perturbing the system, a rapid transition between two attractors was identified in the deterministic case. Further investigation using the stochastic approach confirmed the existence of two attractors in the model. The two attractors symbolize a neuronal- and a non-neuronal cell state. The transition between the two states was identified by perturbing SCP1, which is a component of RESTc. Specifically, it was the perturbation of the PTB binding affinity on SCP1 that caused the rapid transition. From the perturbation studies, we predict that controlling not only REST but also those components recruited by it is important for obtaining a high yield conversion scheme.

In summary, the suggested model is capable of capturing the main features of the experimental time series. Perturbation studies revealed two attractors of the model. These attractors are in agreement with the expected behavior that a cell would have in vitro during neural transdifferentiation. Either a cell is in a non-neuronal state, or in a neuronal state. The model could be expanded to include more nodes. However, this adds complexity during the process of fitting model parameters. Modeling a RESTc consisting of additional components may prove important if the model is to be extended to include a more detailed network. Such an approach could explore the components significance in neuronal reprogramming. For the system we model here, SCP1 and REST seem sufficient to characterize RESTc. With these simplifications in mind, it is pleasing that the model is able to capture the main parts of the transdifferentiation process.

To validate the model, the hypothetical interactions need to be verified. It is possible that the direct connections between genes made here are the result of a longer chain of interactions. An interesting point would be if these hypothetical interactions are indeed part of the cell regulatory circuit. This will be one of the most crucial parts for the regulatory gene network obtained in this study. Verifying the trend seen in the bulk time series with single cell time series is also essential for the regulatory gene network presented here.

Future work with this model involves extending it further, such that it includes the process of generating dopaminergic neurons. Here, the experimental conversion rate is still relatively low. This would further our understanding of the dynamics that occur in the cell, hence facilitating the process of obtaining a high yield dopaminergic conversion scheme. Such a breakthrough would possibly open up new pathways for medical treatments and disease modeling of diseases such as Parkinson's disease.

Appendices

A Implementation details

In both search methods viral Ascl1, see section 3.2, is set as a constant parameter. We choose the value of this parameter (virA in equation (31)) to be 1, representing a normalized level. The best deterministic fit is shown in figure 8, other tested network topologies are shown in appendix C. The simulation time, represented on the x-axis of figure 8, is from $t = 0$ hours to $t = 120$ hours, with data points at $t_p \in \{0, 8, 24, 48, 72, 120\}$. In the simulated annealing approach, parameters are initialized randomly within set ranges, see table 1. The rate equations were then solved using ode45 in MATLAB, a Runge-Kutta 4,5 (RK45) based differential equation solver. In the genetic algorithm approach, parameters are initialized within the 64-bit floating point value. The rate equations were solved using the standard RK4 method.

Simulated annealing

The objective function, also known as the cost function, is an important concept in simulated annealing. This is the function that translates each sampled state to an artificial energy. The cost function is problem specific. The aim of the optimization is to produce time evolutions that matches experimental time series. In order to perform such an optimization, the cost function has to translate the difference between the generated theoretical curves to the data as a single number, the artificial energy. In this work, we use a least square cost function,

$$E = \sum_j \left(\sum_i (x_i^j - x_{i,\text{data}}^j)^2 \right). \tag{32}$$

Here, the outer sum symbolizes each gene j, and the inner sum is over every matching pair i between the generated gene expression x_i^j and data point $x_{i,\text{data}}^j$.

Optimizing the fit to data is equivalent to minimizing equation (32). Since the level of gene expression varies between genes (see figure 6), using a direct least square calculation will result in certain genes contributing more to the overall cost. To avoid this we normalize the data in the simulated annealing approach, dividing every data point with the maximum concentration value measured for that specific gene. The maximum value divided by is chosen from the total time period that we fit to. After transforming the data, each gene has a maximum concentration value of 1. In this way, if the order of magnitude differs a lot between genes, the resulting cost will not be affected substantially. To further facilitate the fitting process we use interpolated data points. These help the algorithm trace data better as deviating solutions are penalized further. In this work we use a total of 5 linearly interpolated points, one between every pair of adjacent data points.

There exists a diverse number of cooling schemes, ranging from monotonically decreasing functions to adaptive non-monotonically decreasing functions. The objective of the cooling schedule is to reduce the temperature quickly while still slow enough such that the algorithm is not trapped in a metastable state. The cooling schedule required is dependent on the specific problem. In this work we use an exponential cooling schedule, which is a common monotonically decreasing cooling schedule,

$$T_n = T_0 \alpha^n. \tag{33}$$

T_0 is the starting temperature and α a parameter. In our simulations $T_0 = 1$, and $\alpha = 0.9$. n denotes the number of temperature decrements. The temperature is updated until a minimum temperature is reached. The state when $T \leq T_{\text{min}}$ was kept as the final value.

The neighborhood or candidate function, the function that samples a new state from the present state, can have a major impact on both the run time and the quality of result. The regulatory gene network consists of 45 parameters in total. These parameters have different degrees of sensitivity associated with them. A neighborhood function should on average produce new states that correspond to small steps in the energy landscape. Otherwise it may be difficult to fine tune the solution toward the global minimum. Thus, we choose a random subset of parameters for updating, drawn with replacement. These parameters are then perturbed around the present value. The function for perturbing a parameter y can be summarized as the following,

$$y_{\text{new}} = \begin{cases} y \cdot \left[1 + a\left(r - \frac{1}{2}\right)\right], & R < 0.5, \\ y \cdot \left[1 + b\left(r - \frac{1}{2}\right)\right], & R \geq 0.5. \end{cases} \tag{34}$$

y_{new} denotes the updated value of parameter y. r and R are uniformly distributed random numbers. The parameters a and b are positive constants that specify the allowed range for the updated parameter. For our simulations $a = 0.2$ and $b = 0.4$. a is smaller than b because this allows the algorithm to take both small and large steps. On average, small perturbations are more likely to occur.

The updating strategy described above is used for all parameters except the Hill coefficients, which are treated as integers,

$$H_{\text{new}} = \begin{cases} H + \text{sgn}\left(\frac{1}{2} - r\right), & R < 0.75, \\ H + 2 \cdot \text{sgn}\left(\frac{1}{2} - r\right), & R \geq 0.75. \end{cases} \tag{35}$$

Here, H_{new} denotes the updated Hill coefficient, and H the current value. sgn represent the sign function, with the properties $\text{sgn}(x) = +1$ for $x > 0$, $\text{sgn}(x) = -1$ for $x < 0$, and 0 otherwise. r and R are uniformly distributed random numbers. Note, again, that the second option corresponds to larger steps in the parameter space. The equations (34) and (35) characterize the core of the neighborhood function. For every type of parameter we also chose an allowed range. Updated parameters that exceed this range are set to the corresponding minimum or maximum value. The set of ranges are shown in table 1.

Table 1: The set of parameter ranges used in the simulated annealing approach.

s_i	[0, 30]
b_i	[0, 1]
H_i	[0, 10]
α_i	[0, 17]
γ_i	[0, 4]
RESTi	[0, 20]
K_d	[0, 2]

Genetic algorithm

An individual in our model is constructed as an ordered list of gene values. The total number of genes is determined by the topology of the regulatory gene network. The winning regulatory gene network consists of 45 genes. A gene in our network is represented using a floating point value representation. The value of a gene is directly read from the genome, except for the Hill parameters, H_i, and the dimerization constant, K_d. These values are chosen as constrained in the ranges $[0, 10]$ respectively $[0, 1]$. A gene x, being a Hill- or dimerization parameter, is transformed when read using,

$$x \to \frac{kx}{1+x}.$$

Here, $k = 10$ for a Hill parameter and $k = 1$ for the dimerization constant.

The genetic algorithm uses a tournament selection that is modified somewhat in comparison to the description of the method given in section 2.2. Here, creating a new generation is performed by breeding and removing a set of individuals and replicating the remaining ones. Breeding a new individual is achieved by choosing two individuals and recombining them. Thus, there exists a possibility for every individual to reproduce. Removing an individual occurs by choosing a subset of 5 individuals from the current population and removing the worst of them. This procedure is repeated four times for every generation. The population size is 100 individuals.

Using the fourth order Runge-Kutta method, the time evolution is determined. Calculating the fitness of an individual is done by summing two terms. The first term of the fitness function stands for the least square error, previously described for the simulated annealing approach (see equation (32)). Here, normalization is performed by taking the difference, $x_i^j - x_{i,\text{data}}^j$, and divide by the corresponding maximum concentration in the set of experimental values, $\{x_{\text{data}}\}_i^j$, for gene j. The resulting value is then squared as usual and added to the overall fitness. The second term is a smoothing regularization term that helps reduce oscillations in the fitted curves by penalizing rapid fluctuations. This is achieved by numerically approximating the integral of the absolute value of the first derivative for each rate equation. Each value calculated is normalized with respect to the corresponding maximum value measured experimentally, similar to the least square method. The size of the penalization is scaled by a regularization parameter such that the smoothing is not too extreme in comparison to the mean squared error of the time evolutions. This regularization parameter is 1/20, determined by hand.

The crossover operation, performed when creating a new individual, is accomplished by randomly combining the genes from each parent. Thus, for each of the 45 genes, the children are given one at random from each parent. This is done in a sequential order such that the position of each gene is conserved.

The mutation operation mutates each gene independently. Here, each gene is multiplied with three independently calculated factors that are either 1 or the exponential of a normally distributed random number. The distributions are centered around 0 with standard deviations log(1.01), log(1.05), and log(1.5). These factors represent respectively very small, small, and large mutations. The probability of each mutation is p, $p/3$, and $p/10$, with $p = 2/45$.

B Complete time series

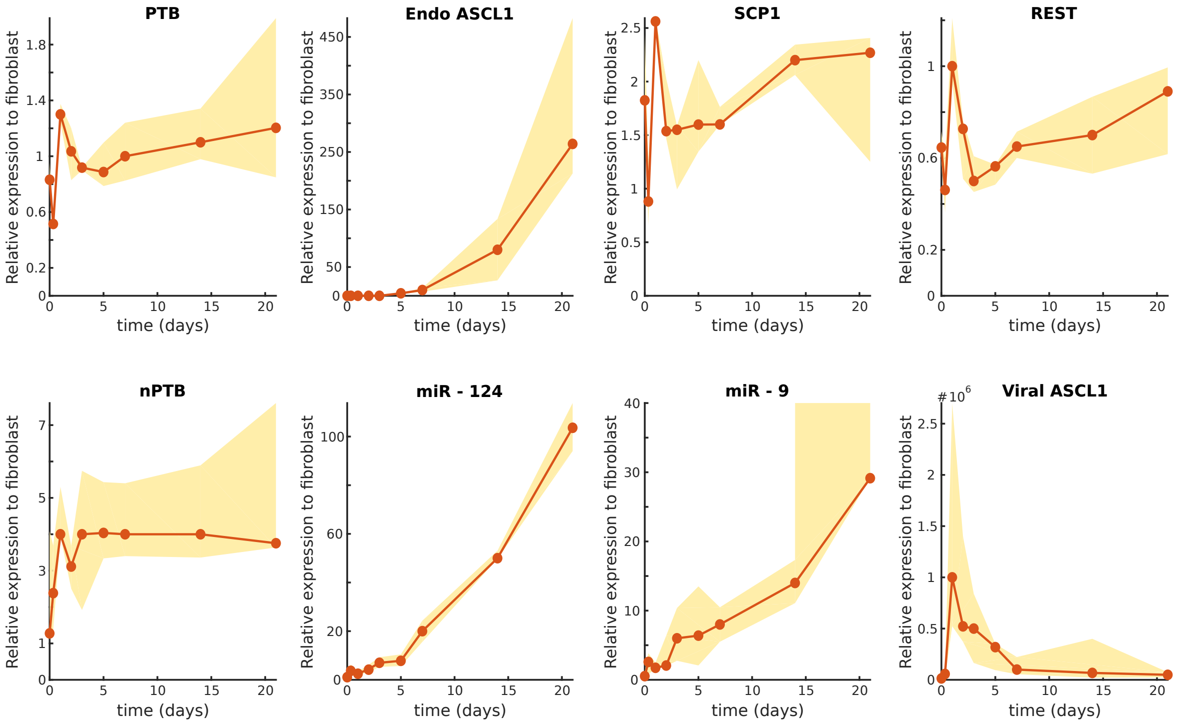
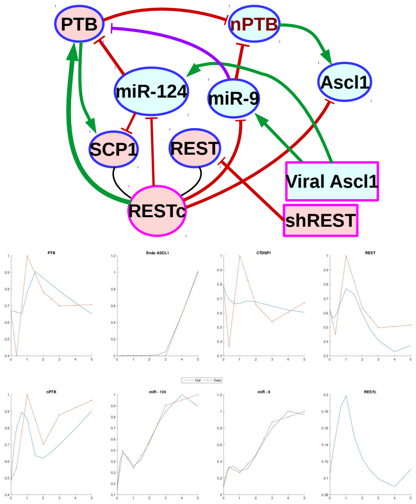
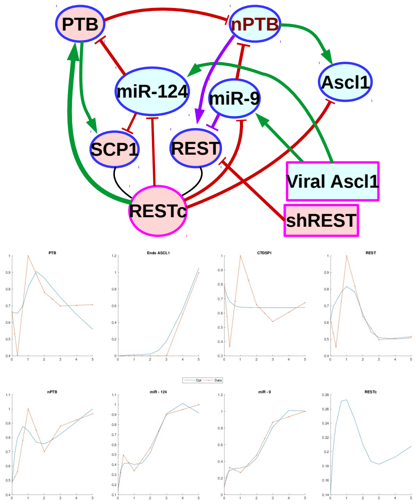


Figure 11: Complete time series data of figure 6 on page 26. Data points are plotted at times $(0, 0.3, 1, 2, 3, 5, 7, 14, 21)$ measured in days. Note the time series of miR-9. Here, the last data point illustrates the minimum value instead of the median value. The reason for this is that the two other data points have extremely high values at roughly $4 \cdot 10^4$ and $2.7 \cdot 10^4$. Since data points are independent of each other, these extreme values may represent time series that are overall much higher in concentration than other time series corresponding to the other data points. For these reasons we neglect plotting the median value at the last time point. Under these assumptions, the dynamics show that after day 5 the cells reach a steady state.

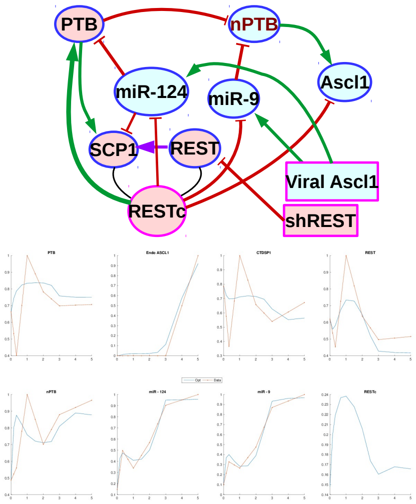
C Network topologies



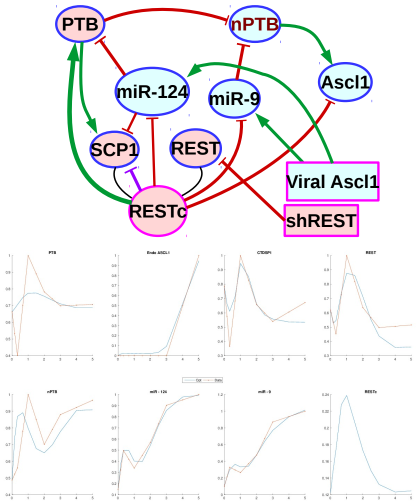
(a)



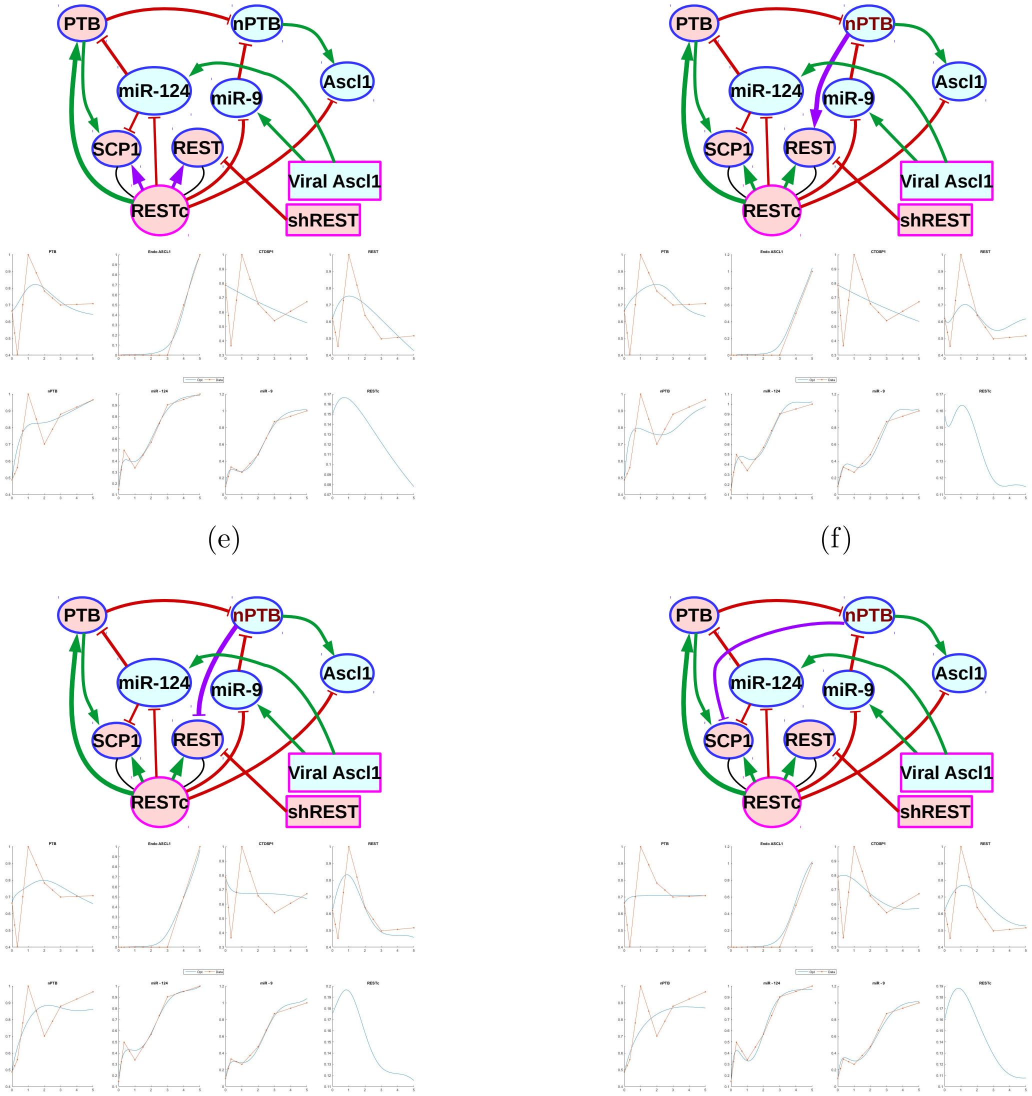
(b)



(c)



(d)



(e)

(f)

(g)

(h)

Figure 12: A set of tested network topologies together with the accompanied fits. Purple arrows indicate which interaction is being tested. Note that the microRNAs 124 and 9 are separated. The hypothetical interaction between RESTc and PTB, as well as nPTB and Ascl1, are present in all simulations. From figure (e) and forward, the activation of SCP1 and REST were adopted as they improved the fitting. The fits shown here were generated using the Simulated Annealing algorithm. The fits are plotted in the order of PTB, Ascl1, SCP1, REST, nPTB, miR-124, miR-9, and RESTc. RESTc is not fitted to experimental data. In the fits CTDSP1 is equivalent to SCP1.

D Parameter set

Table 2: Parameter table of the best parameter values for equation (26)-(31). These values were obtained by minimizing the fitness function of the genetic algorithm. Note that virA is a constant parameter of the model. The resulting time evolutions are shown in figure 8.

(a) Table 1

α_1	0.5508
α_2	5.6378
α_3	2.0636
α_4	1.6195
α_5	1.6073
α_6	0.3443
b_1	0.0011
b_2	$4.4487{\cdot}10^{-6}$
b_3	0.0084
b_4	1.3359
b_5	0.0327
s_1	4.5607
s_2	0.4070
s_3	0.2462
s_4	8.5180
s_5	0.0430
s_6	0.0616
s_7	1.1035
s_8	6.3755
s_9	2.8500
s_{10}	0.3135
s_{11}	1.9862
s_{12}	5.9762
s_{13}	3.1450

(b) Table 2

H_1	1.5806
H_2	$1.1495{\cdot}10^{-5}$
H_3	10.0000
H_4	4.3678
H_5	3.2549
H_6	2.7423
H_7	3.8116
H_8	1.8091
H_9	2.3217
H_{10}	3.3484
H_{11}	5.5800
H_{12}	1.0000
H_{13}	3.0609
γ_1	0.3235
γ_2	$1.2013{\cdot}10^{-6}$
γ_3	0.3502
γ_4	0.9509
γ_5	0.0082
γ_6	0.0291
RESTi	3.3827
K_d	0.7595
virA	1

References

[1] JA Obeso, MC Rodríguez−Oroz, B Benitez−Temino, FJ Blesa, J Guridi, C Marin, and M Rodriguez. Functional organization of the basal ganglia: therapeutic implications for parkinson's disease. *Movement Disorders*, 23:548–559, 2008.

[2] O Lindvall and A Björklund. Cell therapy in parkinson's disease. *NeuroRx*, 1:382–393, 2004.

[3] N M King and J. Perrin. Ethical issues in stem cell research and therapy. *Stem cell research & therapy*, 5:85, 2014.

[4] K Takahashi and S Yamanaka. Induction of pluripotent stem cells from mouse embryonic and adult fibroblast cultures by defined factors. *cell*, 126:663–676, 2006.

[5] K Takahashi, K Tanabe, M Ohnuki, M Narita, T Ichisaka, K Tomoda, and S Yamanaka. Induction of pluripotent stem cells from adult human fibroblasts by defined factors. *cell*, 131(5):861–872, 2007.

[6] J Yu, MA Vodyanik, K Smuga-Otto, J Antosiewicz-Bourget, JL Frane, S Tian, J Nie, GA Jonsdottir, V Ruotti, R Stewart, and II Slukvin. Induced pluripotent stem cell lines derived from human somatic cells. *science*, 318:1917–1920, 2007.

[7] M Wernig, JP Zhao, J Pruszak, E Hedlund, D Fu, F Soldner, V Broccoli, M Constantine-Paton, O Isacson, and R Jaenisch. Neurons derived from reprogrammed fibroblasts functionally integrate into the fetal brain and improve symptoms of rats with parkinson's disease. *Proceedings of the National Academy of Sciences*, 105: 5856–5861, 2008.

[8] J Kim, R Ambasudhan, and S Ding. Direct lineage reprogramming to neural cells. *Current opinion in neurobiology*, 22:778–784, 2012.

[9] T Vierbuchen, A Ostermeier, Z P Pang, Y Kokubu, T C Südhof, and M Wernig. Direct conversion of fibroblasts to functional neurons by defined factors. *Nature*, 463: 1035–1041, 2010.

[10] O L Wapinski, T Vierbuchen, K Qu, Q Y Lee, S Chanda, D R Fuentes, P G Giresi, Y H Ng, S Marro, N F Neff, et al. Hierarchical mechanisms for direct reprogramming of fibroblasts to neurons. *Cell*, 155:621–635, 2013.

[11] U Pfisterer, A Kirkeby, O Torper, J Wood, J Nelander, A Dufour, A Björklund, O Lindvall, J Jakobsson, and M Parmar. Direct conversion of human fibroblasts to dopaminergic neurons. *Proceedings of the National Academy of Sciences*, 108:10343–10348, 2011.

[12] AS Yoo, AX Sun, L Li, A Shcheglovitov, T Portmann, Y Li, C Lee-Messer, RE Dolmetsch, RW Tsien, and GR Crabtree. MicroRNA-mediated conversion of human fibroblasts to neurons. *Nature*, 476:228–231, 2011.

[13] Z P Pang, N Yang, T Vierbuchen, A Ostermeier, D R Fuentes, T Q Yang, A Citri, V Sebastiano, S Marro, T C Südhof, et al. Induction of human neuronal cells by defined transcription factors. *Nature*, 476:220–223, 2011.

[14] R Ambasudhan, M Talantova, R Coleman, X Yuan, S Zhu, S A Lipton, and S Ding. Direct reprogramming of adult human fibroblasts to functional neurons under defined conditions. *Cell stem cell*, 9:113–118, 2011.

[15] V Broccoli, M Caiazzo, and MT Dell'Anno. Setting a highway for converting skin into neurons. *Journal of molecular cell biology*, 3:322–323, 2011.

[16] J Visvanathan, S Lee, B Lee, JW Lee, and SK Lee. The microrna mir-124 antagonizes the anti-neural rest/scp1 pathway during embryonic cns development. *Genes & development*, 21:744–749, 2007.

[17] A N Packer, Y Xing, S Q Harper, L Jones, and B L Davidson. The bifunctional microrna mir-9/mir-9* regulates rest and corest and is downregulated in huntington's disease. *Journal of Neuroscience*, 28:14341–14346, 2008.

[18] D S Johnson, A Mortazavi, R M Myers, and B Wold. Genome-wide mapping of in vivo protein-dna interactions. *Science*, 316:1497–1502, 2007.

[19] N Ballas, C Grunseich, D D Lu, J C Speh, and G Mandel. Rest and its corepressors mediate plasticity of neuronal gene chromatin throughout neurogenesis. *Cell*, 121: 645–657, 2005.

[20] M Yeo, SK Lee, B Lee, E C Ruiz, S L Pfaff, and G N Gill. Small ctd phosphatases function in silencing neuronal gene expression. *Science*, 307:596–600, 2005.

[21] E V Makeyev, J Zhang, M A Carrasco, and T Maniatis. The microRNA miR-124 promotes neuronal differentiation by triggering brain-specific alternative pre-mRNA splicing. *Molecular cell*, 27:435–448, 2007.

[22] G C Coutinho-Mansfield, Y Xue, Y Zhang, and XD Fu. Ptb/nptb switch: a post-transcriptional mechanism for programming neuronal differentiation. *Genes & development*, 21:1573–1577, 2007.

[23] Y Xue, K Ouyang, J Huang, Y Zhou, H Ouyang, H Li, G Wang, Q Wu, C Wei, Y Bi, and L Jiang. Direct conversion of fibroblasts to neurons by reprogramming PTB-regulated microRNA circuits. *Cell*, 152:82–96, 2013.

[24] Y Xue, H Qian, J Hu, B Zhou, Y Zhou, X Hu, A Karakhanyan, Z Pang, and XD Fu. Sequential regulatory loops as key gatekeepers for neuronal reprogramming in human cells. *Nature neuroscience*, 19:807–815, 2016.

[25] J Drouin-Ouellet, S Lau, PL Brattås, DR Ottosson, K Pircs, DA Grassi, LM Collins, R Vuono, A Andersson Sjöland, G Westergren-Thorsson, L Minthon, H Toresson, RA Barker, J Jakobsson, and M Parmar. Rest suppression mediates neural conversion of adult human fibroblasts via microrna-dependent and-independent pathways. *EMBO molecular medicine*, 9:1117–1131, 2017.

[26] D P Bartel. MicroRNAs: target recognition and regulatory functions. *cell*, 136:215–233, 2009.

[27] N Bushati and S M Cohen. microRNA functions. *Annu. Rev. Cell Dev. Biol.*, 23: 175–205, 2007.

[28] D P Bartel. Micrornas: genomics, biogenesis, mechanism, and function. *cell*, 116: 281–297, 2004.

[29] M B Victor, M Richner, T O Hermanstyne, J L Ransdell, C Sobieski, PY Deng, V A Klyachko, J M Nerbonne, and A S Yoo. Generation of human striatal neurons by microrna-dependent direct conversion of fibroblasts. *Neuron*, 84:311–323, 2014.

[30] N Urbán, D LC van den Berg, A Forget, J Andersen, J AA Demmers, C Hunt, O Ayrault, and F Guillemot. Return to quiescence of mouse neural stem cells by degradation of a proactivation protein. *Science*, 353:292–295, 2016.

[31] D S Castro, D Skowronska-Krawczyk, O Armant, I J Donaldson, C Parras, C Hunt, J A Critchley, L Nguyen, A Gossler, B Göttgens, et al. Proneural bhlh and brn proteins coregulate a neurogenic program through cooperative binding to a conserved dna motif. *Developmental cell*, 11:831–844, 2006.

[32] J Eriksson. Simulating direct conversion of adult fibroblasts to neurons, 2017. Student Paper.

[33] P L Boutz, P Stoilov, Q Li, CH Lin, G Chawla, K Ostrow, L Shiue, M Ares, and D L Black. A post-transcriptional regulatory switch in polypyrimidine tract-binding proteins reprograms alternative splicing in developing neurons. *Genes & development*, 21:1636–1652, 2007.

[34] R Spellman, M Llorian, and C WJ Smith. Crossregulation and functional redundancy between the splicing regulator ptb and its paralogs nptb and rod1. *Molecular cell*, 27: 420–434, 2007.

[35] S Goutelle, M Maurin, F Rougier, X Barbaut, L Bourguignon, M Ducher, and P Maire. The hill equation: a review of its capabilities in pharmacological modelling. *Fundamental & clinical pharmacology*, 22:633–648, 2008.

[36] SS Antman JE Marsden, L Sirovich S Wiggins, L Glass, RV Kohn, and SS Sastry. *Interdisciplinary Applied Mathematics*. Springer, 1993.

[37] MA Shea and GK Ackers. The or control system of bacteriophage lambda: A physical-chemical model for gene regulation. *Journal of molecular biology*, 181:211–230, 1985.

[38] J D Watson, F HC Crick, et al. Molecular structure of nucleic acids. *Nature*, 171: 737–738, 1953.

[39] P Nelson, M Radosavljević, and S Bromberg. *Biological physics: Energy, Information, Life*. WH Freeman New York, updated first edition, 2008.

[40] R G Roeder. The role of general initiation factors in transcription by rna polymerase ii. *Trends in biochemical sciences*, 21:327–335, 1996.

[41] S Hahn. Structure and mechanism of the rna polymerase ii transcription machinery. *Nature structural & molecular biology*, 11:394, 2004.

[42] V Olariu, C Lövkvist, and K Sneppen. Nanog, oct4 and tet1 interplay in establishing pluripotency. *Scientific reports*, 6, 2016.

[43] S Kirkpatrick, CD Gelatt, and MP Vecchi. Optimization by simulated annealing. *science*, 220:671–680, 1983.

[44] N Metropolis, AW Rosenbluth, MN Rosenbluth, AH Teller, and E Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21: 1087–1092, 1953.

[45] WH Press, SA Teukolsky, WT Vetterling, and BP Flannery. *Numerical recipes in C*, volume 2. Cambridge university press Cambridge, 1996.

[46] DS Johnson, CR Aragon, LA McGeoch, and C Schevon. Optimization by simulated annealing: an experimental evaluation; part i, graph partitioning. *Operations research*, 37:865–892, 1989.

[47] D Whitley. A genetic algorithm tutorial. *Statistics and computing*, 4:65–85, 1994.

[48] DT Gillespie. Exact stochastic simulation of coupled chemical reactions. *The journal of physical chemistry*, 81:2340–2361, 1977.

[49] DT Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of computational physics*, 22:403–434, 1976.

[50] T Friedmann. A brief history of gene therapy. *Nature genetics*, 2:93, 1992.

[51] C E Thomas, A Ehrhardt, and M A Kay. Progress and problems with the use of viral vectors for gene therapy. *Nature Reviews Genetics*, 4:346, 2003.

[52] S Gascón, G Masserdotti, G L Russo, and M Götz. Direct neuronal reprogramming: achievements, hurdles, and new roads to success. *Cell stem cell*, 21:18–34, 2017.

[53] S C Daubner, T Le, and S Wang. Tyrosine hydroxylase and regulation of dopamine synthesis. *Archives of biochemistry and biophysics*, 508:1–12, 2011.