Aus dem Lehrstuhl für Bioinformatik

der Universität Würzburg

Vorstand: Professor Dr. med. Thomas Dandekar

Control Centrality in Non-Linear Biological Networks

Inaugural - Dissertation

zur Erlangung der Doktorwürde der

Medizinischen Fakultät

der

Julius-Maximilians-Universität Würzburg

vorgelegt von

Stefan Karl

aus Iphofen

Würzburg, Mai 2016

Referent: Prof. Dr. med. Thomas Dandekar

Korreferent: Priv.-Doz. Dr. rer. nat. Dr. med. Christoph U. Schoen

Dekan: Prof. Dr. med. Matthias Frosch

Tag der mündlichen Prüfung: 22. Juni 2017

Der Promovend ist Arzt

Caminante, no hay camino,
sino estelas en la mar.

Table of Contents

1 Introduction

How to treat a disease? This pivotal question of medicine still remains unanswered to a large extent for a multitude of widespread medical conditions.

From the perspective of system biology, many of these afflictions like cardiovascular diseases or cancer can be seen as a dysregulation, i.e. a deviation from normal regulatory processes in the human body. Based on results from molecular biology, medical system biology endeavors thus to understand, and ultimately control, the complex regulatory networks underlying many common ailments, leading to novel pharmacological targets and therapeutic approaches. Even today, gaining control over a systems often equals health for the patient.

To further the understanding of the dynamics of complex biological regulatory systems and how to control them is the aim of this dissertation.

1.1 Genetic regulatory networks

Whether single cells, organisms or whole ecosystems, biological systems are governed by regulatory processes. The number of predators in a habitat is regulated by the available prey in much the same way as the insulin secretion in pancreatic beta cells is regulated by the blood glucose level and other influences.

The mechanism of the regulatory interactions depends on the biological system. While intracellular regulation is usually based on the control of gene expression and protein activity, at the organism level hormonal, nerval and metabolic control are prevalent.

In a localized view, these regulations appear as linear or tree-like signaling cascades, but a more comprehensive scope very often reveals feedback loops and interactions with other systems giving rise to complex regulatory networks. These feedbacks, like the reduction of blood sugar levels in peripheral cells by insulin effects, are the foundation of

homeostasis and equilibrium in biological cells, as well as of rapid adaptation to external influences.

The sum of all regulatory processes inside a cell determines its internal state as well as its phenotype and external behavior. The single cell then interacts with structural factors such as diffusion compartments forming organism-level regulatory networks.

1.2 Regulatory interactions inside of cells

The chapter provides a brief summary of the most prevalent regulatory interactions inside of cells. For a more detailed description, which goes beyond the scope of this introductory section, see the citations.

1.2.1 Transcriptional regulation of gene expression

Among the most important targets of regulatory elements in prokaryotic and eukaryotic cells are proteins, which in turn have catalytic functions as enzymes, structural functions as in the cytoskeleton and regulatory functions as transcription factors and in protein-protein interactions.

The first step where the activity of a protein may be controlled is during its formation when DNA is transcribed to pre-mRNA, either at the initiation of this process, or during the elongation of the pre-mRNA transcript.

A typical transcription factor (TF) regulates transcription by binding to specific DNA-sequences in cis-regulatory-modules (CRM) close to the target gene, looping the DNA between the CRM and the promoter element where the transcription begins, and regulating the attraction of RNA-polymerases and coactivators to this transcription initiation complex. CRMs can act as enhancers, which increase the gene transcription, or silencers which decrease it, thus distinguishing activating TFs (activators) from inhibiting TFs (repressors) (Lee, Young 2013; Lelli et al. 2012) depending on their CRM-specificity. Repressors may also inhibit DNA transcription by blocking the access to

enhancers, sometimes forming large multi-protein repressors complexes (Adachi, Monteggia 2014).

While in prokaryotes a gene is typically regulated by a single TF, in eukaryotes multiple TFs share in control of a gene. In both domains, a TF potentially regulates a large number of genes (Lee, Young 2013; Lelli et al. 2012).

Similar to TFs, transcription is also modulated by other proteins such as chromatin regulators which exert influence by altering the accessibility of the DNA strand (Di Croce, Helin 2013), and by regulators of transcriptional elongation (Kwak, Lis 2013).

Non-protein regulators include nucleotides such as long non-coding RNA (lncRNA) (Orkin, Hochedlinger 2011), which act for example by recruiting repressor proteins to one or multiple DNA sites (Nagano, Fraser 2011).

After transcription, the pre-mRNA is subject to several processes such as splicing, end modifications, and transport, before it becomes mature mRNA and is translated into a protein. Translation may be inhibited by small RNA molecules (miRNA) which also contribute to mRNA degradation (Lee, Young 2013).

1.2.2 Regulation of protein activity.

Once a protein has been formed, its activity can still be modulated.

Allosteric control by proteins and other molecules stabilizes the structure of a target protein such an ion channel in an activated or inhibited conformation (Bertrand, Gopalakrishnan 2007).

Another type of regulation occurs in G proteins, where ligand-induced conformation changes in G protein-coupled receptors secondarily lead to the exchange of GDP by GTP in the G protein, thereby inducing the dissociation of its active G_α subunit. The now active subunit then induces signaling pathways such as the synthesis of the second messenger cyclic adenosine monophosphate (cAMP) by adenylyl cyclase (Ritter, Hall 2009).

Proteins can also be regulated by reversible covalent modifications inducing conformation changes, like phosphorylation by protein kinases such as cAMP-activated Protein kinase A (Taylor et al. 2004).

1.2.3 Integration of regulatory influences

In numerous types of regulations, multiple regulators share control of a regulatory target. In transcriptional regulation, transcription factors and cofactors can form enhanceosomes, i.e. large protein complexes with altered DNA-binding affinity and DNA-binding specificity (Lelli et al. 2012). For example, in the well-studied interferon-β CRM, eight proteins cooperatively binding to the enhancer are necessary for the expression of interferon-β (Panne 2008). The activity of the enhanceosomes is modulated by repressors and other silencing mechanism mentioned in 1.2.1.

Many biochemical processes thus integrate influences from multiples sources, giving rise to complex synergistic and antagonistic interactions between regulators.

1.2.4 Autoregulation

In many biological systems, transcription factors and other regulatory elements influence their own activation level. For example, TFs can upregulate their own promoter, leading to autoregulatory loops essential for the maintenance of cell states (Young 2011; Mendoza, Pardo 2010). Autoinhibition (Lelli et al. 2012), on the other hand, contributes to homeostasis of regulation and cell functions (Pufall, Graves 2002).

1.2.5 Decay

In many of the aforementioned regulatory mechanisms, the activated component is subject to spontaneous downregulation unless permanent activating influences are present. For example, active proteins are degraded in proteasomes (Lecker et al. 2006), mRNA by exonucleases (Garneau et al. 2007), phosphorylation is reversed by phosphatases (Cheng et al. 2011), cAMP is degraded by phosphodiesterases (Omori,

Kotera 2007), and GTP in activated G proteins is hydrolyzed back to GDP by an intrinsic GTPase-activity of the G_α subunit (Ritter, Hall 2009).

These decays enable the termination of regulatory influences and thus contribute to the homeostasis of signaling processes.

1.2.6 Interaction databases

Numerous publicly available databases collect information on regulatory processes in different organisms.

TFCat, a curated database of transcription factors based on experimental evidence, lists 665 genes which code for TFs in either human, mouse or both as of 09/2014 (Fulton et al. 2009). A manual survey of DNA-bindings domains in the human genome for which there is experimental evidence for regulatory functions or which exhibit an equivalent protein domain arrangement to known TFs found 1,391 probable TFs (Vaquerizas et al. 2009). The HTRIdb database aggregates 51,871 TF-target-gene interactions between 284 TFs and 18,302 target genes (Bovolenta et al. 2012).

Experimentally confirmed protein-protein-interaction are curated in the BioGRID database, which lists 173,700 non-redundant physical (i.e. non-genetic) interactions in humans (Chatr-Aryamontri et al. 2013).

Combining transcriptional and post-transcriptional regulation with direct protein-protein interactions, the STRING database contains 332,235,675 confirmed and predicted protein-protein interactions of 5,214,234 proteins in 1,133 organisms as of version 9.1 (12/2014) (Franceschini et al. 2013).

As to miRNA regulation, the miRBase database contains 28,645 miRNA loci in 223 species as of version 21 (06/2014), of which 1,881 are high confidence miRNA in humans (Kozomara, Griffiths-Jones 2014).

1.2.7 Network databases

Almost all cell parts thus participate in regulatory processes as regulators or targets. If the targets have regulatory functions themselves, trees of signaling cascades are formed, and with the integration of several pathways and feedback loops, regulatory networks.

While databases containing single regulations and associations have reached high levels of maturity due to an abundance of data from automated high-throughput analyses, combining these elements into high-quality regulatory networks which describe a signaling process in biological systems is very often still a manual task, for example due to a lack of quantitative data regarding the strength of the interactions.

Those manually assembled networks are then usually presented in single publications. Nevertheless, more and more networks are included in signaling pathway databases such as KEGG PATHWAY (Kanehisa et al. 2012) and SPIKE (Paz et al. 2011).

1.3 Mathematical modeling of regulatory networks

Before a regulatory network can be analyzed on a computer, it has to be transformed into a mathematical model. The first step in this modeling process is usually to display the biological interactions in a network graph.

1.3.1 Network graphs

A directed graph consists of a vector of graph nodes $(x_1,...,x_n)$ and a set of edges $E = \{x_{i_1} \to x_{j_1},...,x_{i_n} \to x_{j_n}\}$.

In regulatory network graphs (Figure 1), the nodes (circles) represent biological entities that can assume different levels of activation, for example genes with changing expression levels, proteins controlled by post-translational modifications, or messenger molecules subject to concentration changes, such as hormones or intracellular transmitters.

Graph edges (arrows) correspond to interactions between the elements of the network, such as transcriptional control, producer-product relationships or allosteric regulation. If an arrow A → B exists in a network graph, i.e. node B is regulated by node A in some

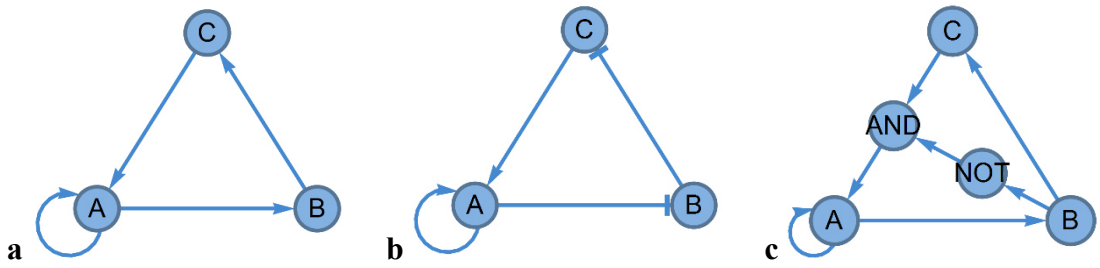


Figure 1: Network graph notation
a A regulatory network with three nodes A, B and C, where the node A influences itself and node B, B influences C and C influences A. In a biological network, A could be a gene that enhances its own expression and the expression of another gene B. **b** An activator-inhibitor network where A has a self-amplifying loop and inhibits B, B inhibits C and C activates A. Normal arrows represent activating influences in the system, and T-shaped arrow represent inhibiting influences. **c** Graph representation of a Boolean network. If multiple arrows lead to the same node, an implicit OR operator is assumed, such that the Boolean functions in this network are $B_A(A,B,C)$ = A OR (C AND (NOT B)), $B_B(A)$ = A and $B_C(B)$ = B.

way, A is called an input node or input to B. The number of arrows originating in a node are called its out-degree, the number of arrows leading to a node is its in-degree.

The sum of the out- and in-degree is the degree of the node. Nodes with high degrees are called hub nodes.

1.3.2 Node values

In the mathematical model, the activation of a biological entity (e.g. a protein) is represented by a numerical value assigned to the corresponding node. Very often, a value

of 1.0 means the node is fully activated, for example a gene at the maximal expression level, and 0.0 means the node is inactive.

In so-called discrete mathematical models, only a finite number of node activations values is possible, usually the values 0 and 1. In continuous models, all values in the interval $[0,1]$ can be assumed by the nodes (Wittmann et al. 2009; Mendoza, Xenarios 2006).

1.3.3 Updating schemes and functions

To proceed from these static network descriptions to dynamic networks which can be simulated on a computer, the values of the nodes need to be linked by updating schemes or functions which describe how the values of the nodes change over time.

If t is the time index of a simulation of a regulatory system, $x_i(t)$ is the value of the node x_i at this time. The time index is usually a dimensionless variable, i.e. it has no unit like seconds or minutes.

The state vector $\boldsymbol{x}(t) = (x_1(t), ..., x_n(t))$ combines the values of all nodes and therefore represents the state of the whole network at time index t. The state $\boldsymbol{x}(0)$ of the network at time index 0, when the simulation is started, is called the initial state.

The simulation then depends on the type of the network.

In discrete models, the simulation proceeds in single steps. In each step, the value of one or more nodes are changed according to updating functions $x_i(t+1) = f_i(x_{i,1}, ..., x_{i,m})$ which determine the new value $x_i(t+1)$ of the node x_i as a function of its input nodes $x_{i,1}, ..., x_{i,m}$. For example, in the network from Figure 1a the value of the node A would be updated according to an updating function $f_A(A, C)$, since the new value of A depends on the nodes A and C.

The behavior of the network also depends on the order in which the updating functions are applied in each step, i.e. the updating scheme. The most common schemes are Classical Random Boolean Networks (CRBN), where all nodes are updated

simultaneously, and Asynchronous Random Boolean Networks (ARBN) where in each step a random node is updated (Gershenson 2004).

In continuous models, the values of the nodes change according to differential equations $\dot{x}_i(t) = f_i(x_{i,1},...,x_{i,m})$, which are simulated starting from the initial state $\boldsymbol{x}(0)$ (Wittmann et al. 2009; Mendoza, Xenarios 2006). A comprehensive overview over differential equations is included in textbooks such as (Bender, Orszag 1999).

1.3.4 Discrete activator-inhibitor networks

A simple modeling concept only distinguishes activating from inhibiting influences. Inhibiting edges are displayed differently from activating edges, for example with T-shaped heads (Figure 1b).

The interplay between activating and inhibiting influences can be defined in different ways. A commonly used concept in discrete networks is that a target node is activated if at least one of the activating influences is present, and none of the inhibiting influences (Wittmann et al. 2009). For example, in the network from Figure 1b, node A would be activated in step $t+1$ if either A or C are activated in step t.

1.3.5 Continuous activator-inhibitor networks

With respect to the structure of differential equations in continuous models, it is possible to distinguish simple, linear equations of the form $\dot{x}_i(t) = \sum_j a_j^i x_j(t) + u_i(t)$ from non-linear equations containing polynomial or exponential expressions. These non-linear components are indispensable for the modeling of switch-like characteristics, where the value of a target node changes sharply if the value of an influencing node rises beyond a certain threshold, and other complex node interactions.

1.3.5.1 The SQDS model

One commonly used non-linear model are Standardized Qualitative Dynamical Systems (SQDS) (Mendoza, Xenarios 2006) which define the differential equation of a node x_i based on the values of its activators x_j^{α} and the values of its inhibitors x_j^{β} as

$$\dot{x}_i = \overbrace{\frac{-e^{0.5h} + e^{-h(\omega_i - 0.5)}}{\left(1 - e^{0.5h}\right)\left(1 + e^{-h(\omega_i - 0.5)}\right)}}^{(*)} - \overbrace{\gamma_i x_i}^{(**)} \text{ where}$$

$$\omega_i = \left(\frac{1 + \sum_j \alpha_j}{\sum_j \alpha_j}\right)\left(\frac{\sum_j \alpha_j x_j^{\alpha}}{1 + \sum_j \alpha_j x_j^{\alpha}}\right)\left(1 - \left(\frac{1 + \sum_j \beta_j}{\sum_j \beta_j}\right)\left(\frac{\sum_j \beta_j x_j^{\beta}}{1 + \sum_j \beta_j x_j^{\beta}}\right)\right)$$

if x_i has activating and inhibiting influences,

$$\omega_i = \left(\frac{1 + \sum_j \alpha_j}{\sum_j \alpha_j}\right)\left(\frac{\sum_j \alpha_j x_j^{\alpha}}{1 + \sum_j \alpha_j x_j^{\alpha}}\right)$$

if x_i has only activating influences, and

$$\omega_i = 1 - \left(\frac{1 + \sum_j \beta_j}{\sum_j \beta_j}\right)\left(\frac{\sum_j \beta_j x_j^{\beta}}{1 + \sum_j \beta_j x_j^{\beta}}\right)$$

if x_i has only inhibiting influences.

In these equations, α_j is the relative strength of the activating influence x_j^{α}, and β_j of the inhibiting influence x_j^{β}. Calculation of ω_i integrates all activating and inhibiting influences in a value between 0 and 1, the (*) part of the formula applies a step function such that the influence of ω_i changes sharply around the threshold 0.5, implementing a switch-like behavior, with the steepness of the step given by the parameter h (Figure 2a).

The (**) parts adds a decay component to the differential equation, with the magnitude of the decay given by γ_i.

The first implementation of the SQDS method was the SQUAD simulation package (Di Cara et al. 2007).

1.3.5.2 Linear networks

In a much simpler linear model the differential equations in an activator-inhibitor network are $\dot{x}_i(t) = \sum_j a_j^i x_j(t) + u_i(t)$ (Liu et al. 2011), i.e. the influence a node x_i receives from a node x_j is proportional to a_j^i, and each node has a time dependent driving influence $u_i(t)$ that represents external stimuli. a_j^i is positive for activating influences, and negative for inhibiting ones. The results of a simulation then not only depends on the initial vector $\boldsymbol{x}(0)$, but also on the driving functions $u_i(t)$.

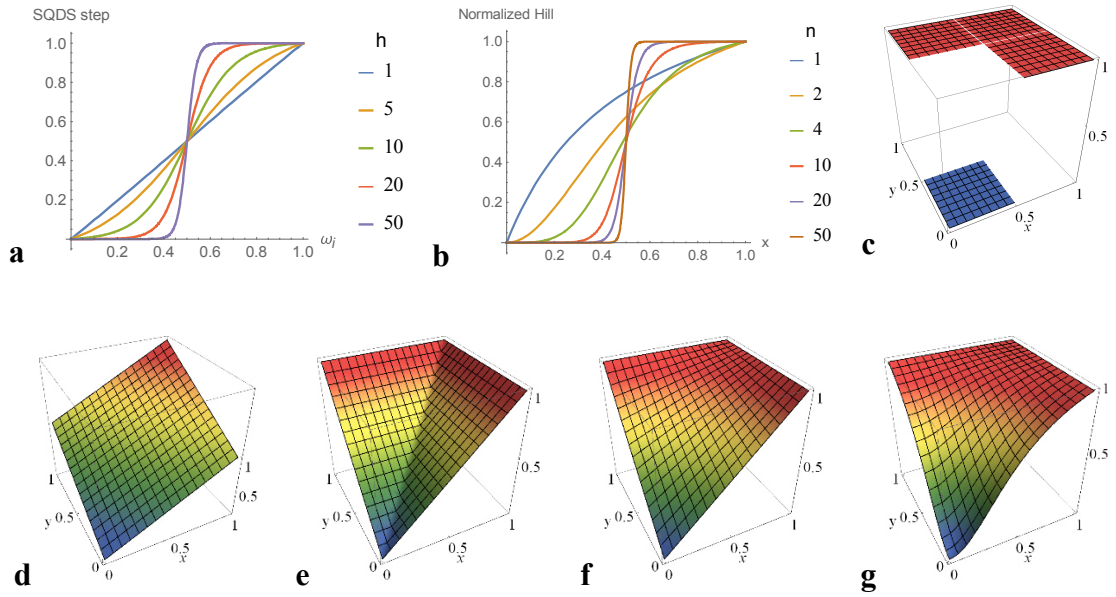


Figure 2: Step functions and interpolations
a Step function of the SQDS model. A node is activated if the integration of the incoming influences leads to a value above 0.5, with the steepness of the response given by the parameter h. **b** Step functions of the normalized HillCube model. For steepness values n over 10 a step similar to the SQDS function is achieved. **c-g** Interpolation of the Boolean function x OR y according to **c** Piecewise linear functions **d** Product-sum fuzzy logic **e** Min-max fuzzy logic **f** BooleCubes **g** Normalized HillCubes

1.3.6 Discrete Boolean networks

As mentioned in section 1.2.3, nodes in biological networks are often controlled by more than one regulator. For example, in typical transcriptional control, multiple transcription factors and corresponding coactivators have to be active at the same time to activate the target, which is called cooperativity (Lelli et al. 2012).

This and other naturally occurring types of regulation cannot be modeled with activator-inhibitor networks, since they implicitly assume that only one the activators must be active to activate the target node.

A solution to this problem is to use Boolean functions as updating functions in discrete models. A Boolean function $f : \mathbf{B}^n \to \mathbf{B}$ maps vectors of binary values $\mathbf{B}^n = \{0,1\}^n$ to either 0 or 1. Just like arithmetic functions can be described using arithmetic operators like "·" or "/", Boolean functions can be written using Boolean operators such as a AND b, which returns 1 if a and b are 1 (and 0 otherwise), a OR b, which returns 1 if a or b are 1, and NOT a, which returns 1 if a is 0.

For example, the Boolean function $B(a,b,c) = a$ AND $(b$ OR NOT $c)$ is 1 if and only if $a = 1$, and either $b = 1$ or $c = 0$.

It can be proven that any possible Boolean function can be expressed using only AND, OR and NOT (Ben-Ari 2012, p. 28). (Ben-Ari 2012) also provides a thorough introduction to Boolean functions in general.

For instance, in a network where two nodes A and B must be active at the same time to activate node C, the updating function would be $B_C(x_A, x_B) = x_A$ AND x_B.

Boolean functions can also be included in the network graph by representing the Boolean operators as network nodes (Figure 1c).

1.3.7 Continuous Boolean networks

Interpolation methods can be used to transfers the versatility of Boolean functions to continuous models, eliminating the shortcomings of activator-inhibitor networks. That is,

for a Boolean function $B:\{0,1\}^n \to \{0,1\}$ a real-valued function $[0,1]^n \to [0,1]$ is defined that extrapolates the behavior of B into the domain $[0,1]^n$.

For example, the function $B(a,b) = a$ AND b returns 1 if $a = 1$ and $b = 1$, and 0 if $a = 1$ and $b = 0$. An interpolation $B':[0,1]^2 \to [0,1]$ of f would therefore be expected to return an intermediate value such as 0.5 for the input $a = 1$ and $b = 0.5$.

(Wittmann et al. 2009) reviewed in detail several common interpolation strategies such as min-max fuzzy logic, product-sum fuzzy logic and piecewise linear functions and found that the resulting interpolations (Figure 2c-e) are either not smooth or do not adequately reproduce the Boolean functions they should interpolate. In response, they introduced the minimal degree polynomial BooleCube interpolation which is smooth and reproduces the Boolean function for all input vectors in $\{0,1\}^n$ (Figure 2e,f).

For a Boolean function $B(x_1,\ldots,x_n)$, the BooleCube interpolation is

$$C[B(x_1,\ldots,x_n)] = \sum_{\bar{x}_1=0}^{1}\sum_{\bar{x}_2=0}^{1}\ldots\sum_{\bar{x}_n=0}^{1}\left(B(\bar{x}_1,\ldots,\bar{x}_n)\prod_{i=1}^{n}(x_i\bar{x}_i + (1-x_i)(1-\bar{x}_i))\right)$$

As an example consider the Boolean function $B'(a,b) = a$ OR b. The BooleCube interpolation is $C[B'] = a + b - ab$ which satisfies $C[B'(a,b)] = B'(a,b)$ for all $(a,b) \in \{0,1\}^2$ and is smooth (Figure 2f).

Wittmann et al. also extended this formalism to include a step function similar to the one found in the SQDS model by modifying the inputs to the BooleCube interpolation by a sigmoid-shaped Hill function $f(x) = x^n / \left(x^n + k^n\right)$ leading to HillCubes and, with a normalized sigmoid function $f(x) = \left(x^n / \left(x^n + k^n\right)\right) / \left(1 / \left(1 + k^n\right)\right)$ (Figure 2b), to normalized HillCubes (Figure 2g).

Note that while in this model the step function modifies the inputs to the influence integration function, in SQDS models the output of the integration function is affected.

Based on the interpolation, the differential equations in BooleCube networks then are $\dot{x}_i = C[B_i(...)]$ where $B_i(...)$ is the Boolean updating function of the node x_i. If the inputs

to the interpolation function are modified with the step functions above, HillCube and normalized HillCube models are formed.

1.4 Network convergence and stable states

1.4.1 Stable states

Equilibria, i.e. situations where the state of the network is does not change anymore at least temporarily, are prevalent in biological systems. These stable network states, where the values of the nodes are constant unless external stimuli influencing the network are added or removed, can usually be related to cell states and cell phenotypes (Espinosa-Soto et al. 2004; Mendoza, Pardo 2010).

In discrete regulatory networks with updating functions $x_i(t+1) = f_i(x_{i,1},...,x_{i,m})$, a state is stable if $x_i = f_i(x_{i,1},...,x_{i,m})$ holds for all nodes x_i (Garg et al. 2008). In other words, all updating functions must evaluate to the value which their target node already holds.

The corresponding condition in continuous networks where $\dot{x}_i = f_i(x_{i,1},...,x_{i,m})$ trivially is $f_i(x_{i,1},...,x_{i,m}) = 0$, i.e. the derivatives of the node values are 0.

1.4.2 Convergence and basins of attraction

If a network is simulated for a very long time, i.e. for time indices $t \to \infty$, two possible scenarios are conceivable. On the one hand, the network state could converge to a situation where the values are no longer changing, i.e. a stable state. On the other hand, node values could keep changing forever, either in a random manner or in an orderly oscillating one. In the usual mathematical models such as SQDS, BooleCube and HillCube, non-converging situations are rare exceptions.

Stable states in discrete as well as continuous models therefore have basins of attractions, i.e. sets of initial values which converge to the stable states for time indices $t \to \infty$. If the

basin of attraction is larger, the stable state has greater robustness against noise in the signaling process (see 3.3.2.2)

1.5 Random networks

While biological networks provide information on real-world regulatory processes, analysis of random networks can elucidate general regulatory mechanism such as the importance of hub nodes and other network motifs.

Two important paradigms for random networks have been proposed (Liu et al. 2011).

In Erdős–Rényi (ER) networks, connections between nodes are set with equal probability, i.e. the arrows in the network graph are distributed randomly over the network (Erdős, Rényi 1959). This leads to rather uniform node degree distributions.

Scale-free (SF) networks on the other hand are constructed such that node degree distributions follow a power law, i.e. the number of network nodes with k connections to other nodes is proportional to $k^{-\lambda}$ where λ is a constant. In the Barabási–Albert network generation model which will be used in this thesis for optimal comparison to previous studies, λ is close to 3 (Barabási, Albert 1999). In scale-free networks, a large majority of the network nodes has a low degree, whereas few hub nodes exhibit high degrees.

Many naturally occurring networks are scale-free (Clauset et al. 2009; Barabási 2003; Albert 2002).

1.6 Network simulation software

1.6.1 Computational complexity

In computer science, the concept of complexity describes how difficult an algorithm is to compute as a function of the length of the input n. Time complexity describes how much time the algorithm takes, and space complexity how much memory it will need.

Both complexities are usually stated using Landau symbols, with the most important one being O. An algorithm with a time complexity in $O(f(n))$ can be computed in a time proportional to $f(n)$ (Wagner 2003).

For example, an algorithm with a time complexity in $O(n)$ would run in a time proportional to the length of the input, and an algorithm in $O(1)$ would run in a time independent of the length of the input. Algorithms with time complexities of $O(2^n)$ or above are usually considered inefficient and not computationally feasible.

Analogously, O can be used for space complexity.

1.6.2 Current implementations

Several of the network types presented above have been implemented in publicly available simulation software.

Discrete Boolean networks and the piecewise linear functions interpolation of Boolean functions are included in the Python library Booleannet (Albert et al. 2008).

Until the development of the Jimena package used in this thesis, the more sophisticated BooleCube and HillCube networks were implemented exclusively in the Matlab package Odefy (Krumsiek et al. 2010). Unfortunately, Odefy currently (08/2015) uses a multidimensional array implementation to store the Boolean functions. That is, for every Boolean function $B(x_1,\ldots,x_n)$ in the network, the program calculates and stores the result for all 2^n possible inputs even before any calculation can be done. The space and time complexity of loading a function B in Odefy is therefore at least $O(2^n)$ where n is the arity of the function. In practice, Odefy crashes reproducibly for functions with arities larger than 9, such that networks where at least one node has more than 9 inputs cannot be simulated by Odefy. This represents a significant limitation since many important hub nodes in biological networks have more than 15 inputs (Young 2011; Ng, Surani 2011).

Odefy's implementation of the BooleCube interpolation is equally inefficient, since it directly calculates the interpolation term for $C[B(x_1,\ldots,x_n)]$ given in section 1.3.7. As

the number of summands in this term is $O\left(2^n\right)$, the total time complexity of the evaluation of a function with n input cannot be lower than $O\left(2^n\right)$. Odefy can therefore not simulate networks with nodes with more than a handful inputs.

The SQDS model for activator-inhibitor networks is implemented efficiently in the SQUAD simulation package (Di Cara et al. 2007).

1.7 Control in regulatory networks

1.7.1 Controllability

In order to derive biological insights from mathematical models, understanding the functions of nodes, interactions and pathways is crucial.

(Liu et al. 2011) introduced controllability in linear networks as the ability of a subset of the nodes to control the behavior of the network insofar as they allow for the steering of the network to any desired state in finite time (Kalman's controllability rank condition) (Kalman 1963). Since they found that the search for driver nodes according to this definition is not computationally feasible for most networks, the concept was weakened to structural controllability (Liu et al. 2011) which approximates the driver nodes at the cost of disregarding numerical parameters describing the interaction of nodes, such as thresholds and relative strengths.

According to structural controllability, in both random and real world networks driver nodes are rarely high degree hub nodes. Around 80% (Liu et al. 2011) of the network nodes are necessary to drive a biological system, a result which has been criticized for contradicting experimental findings suggesting that a small number of nodes dominate network behavior (Müller, Schuppert 2011; Ng, Surani 2011; Lee, Young 2013).

In a similar approach, (Cornelius et al. 2013) investigated how networks can be controlled by subsets of the nodes whose activity can be reduced at will at any given time, but not increased. This limitation to deactivating manipulations lacks biological plausibility since nodes in real-world systems can be subject to unphysiological activation as well as

inhibition, as commonly observed for overexpressed oncogenes or targets of pharmacological interventions.

1.7.2 Control centrality

While controllability deals with the network as a whole, control centrality is a feature of a single network node.

The notion of centrality, i.e. the "importance" of a node in a network, has been defined for regulatory and non-regulatory networks by different approaches such as eigenvector centrality (Bonacich, Lloyd 2001), Page-Rank (Page et al. 1999), closeness and betweenness centrality (Freeman 1978), routing centrality (Dolev et al. 2010), minimum dominating sets (Nacher, Akutsu 2013) and so on. In biological systems, the most important concept is control centrality, i.e. the ability of a node to control the system.

A definition for control centrality derived from structural controllability (Liu et al. 2012) suffers from the same plausibility and applicability problems, such as considerably underestimating the amount of network control that can be concentrated in a single node and being limited to linear networks.

Based on their notion of successive deactivating influences, (Cornelius et al. 2013) also introduced a control centrality concept they call "participation rate" which inherits many plausibility problems. For example, in a network describing T cell survival in T cell large granular lymphocyte leukemia (T-LGL) discussed in the original article, they fail to identify interleukin 15 (IL-15) and platelet-derived growth factor (PDGF), which are sufficient to reproduce all known dysregulations in T-LGL (Leblanc et al. 2012; Zhang et al. 2008), as essential regulators.

1.7.3 Cooperativity

Beyond direct interactions found in single regulations (1.2.3), nodes in regulatory networks can have functionally synergistic or antagonistic effects even over long network distances. For example, the insulin secretion of human pancreatic beta cells is regulated

mainly by the blood glucose level leading to the depolarization of beta cells, but modulated by a number of hormones and neurotransmitters. Synergists of blood glucose like the gut hormone glucagon-like peptide 1 (GLP-1) increase insulin secretion by influencing membrane depolarization and insulin exocytosis, while antagonists like somatostatin ultimately lead to membrane hyperpolarization and thus inhibit insulin secretion (Rorsman, Braun 2013). Although these regulators never interact directly, they still act in synergistic and antagonistic ways on the network.

Network-wide cooperativity effects like these cannot be captured by previous approaches to control in regulatory networks.

1.8 Motivation and aim of this thesis

The aim of this thesis is therefore to overcome the shortcomings in the current understanding of control in regulatory networks by establishing a new set of control centrality metrics which reproduce experimentally known node functions, leading to new findings regarding cooperativity, network robustness, controllability, and signaling processes.

Although several results are obtained for plant and animal networks (*Arabidopsis thaliana* and *Mus musculus*), this medical thesis mainly focusses on regulatory networks in humans. For example, analysis of a network regarding chondrocyte proliferation demonstrates how the novel metrics elucidate medically relevant oncogenes and pharmacological targets, while an extensive study of a model of human T-helper cell differentiation directly relates functions of the human immune systems to mathematical properties of network nodes and pathways.

To compute the new centrality metrics efficiently and accurately, a new simulation framework without the limitations of previous software (1.6) had to be developed.

Note that in some details the structure of this thesis deviates from a strict separation of methods, results and discussion. Since a new set of methods is established for the first

time, it seems expedient to introduce all new definitions and algorithms thoroughly in the Methods chapter. In the Results chapter, those methods are applied to a number of practical examples, each including a short introduction, an explanation of the methods specific to the example, and a succinct discussion of the respective results. The Discussion chapter then integrates all results into a broader view.

2 Methods

This chapter first describes Jimena, a new network simulation framework developed by me, and what sets it apart from preexisting software solutions. The second part then details the new control centrality metrics and related algorithms implemented in Jimena.

2.1 The Jimena simulation package

To compute the new control centralities, it is necessary to perform large numbers of network simulations and to dynamically change the network topology during simulations.

As discussed in section 1.6.2, the more versatile BooleCube and HillCube networks cannot be simulated in Odefy, their only implementation, if they are even moderately complex. In both Odefy and SQUAD, dynamic network changes are not possible.

To tackle these computational complexity issues and to gain full control over the simulation process, Jimena was created.

2.1.1 Implementation

Jimena was developed as a platform independent Java application and an attached, but independently usable, Java library. The software and its source code, as well as manuals and example networks, are publicly available (Karl 2013) under the GNU Lesser General Public License.

Figure 3: The Jimena simulation framework

a An activator-inhibitor network in yEd, ready to be imported into Jimena. Arrows with a diamond-shaped arrow head are inhibiting.

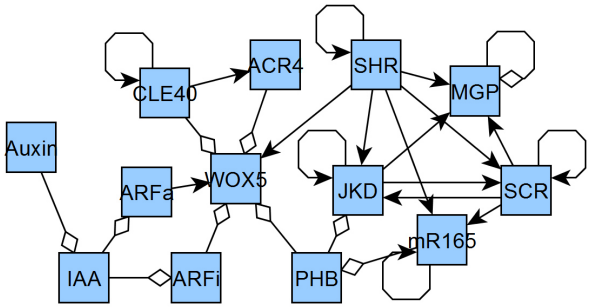
b The initial values of the nodes and others parameters describing the regulatory network such as decays, steepnesses and relative strengths can be set.

c Perturbations can be added to the network nodes. A perturbation controls the value of a single network node during the simulation, such that, for example, the reaction of the network to external stimuli can be studied.

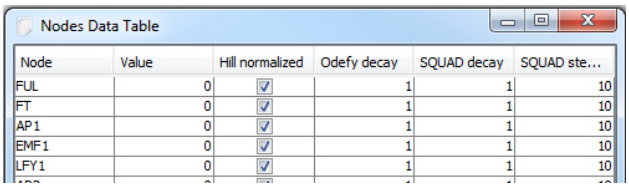
d After the network has been simulated, time series data from the simulation can be displayed and exported.

e Stable states can be searched and exported.

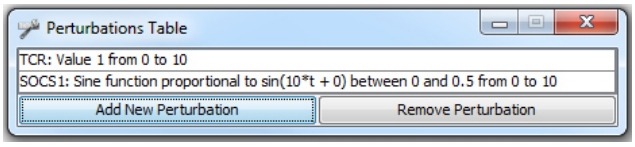
a



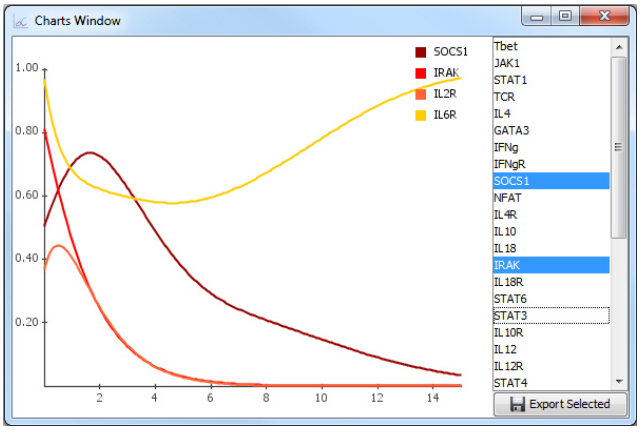
b



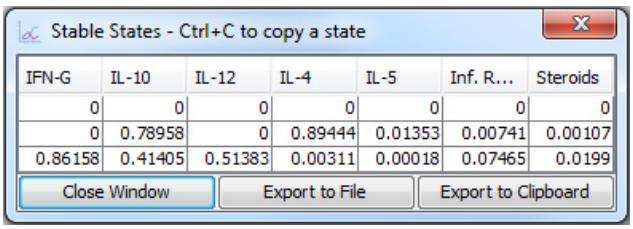
c



d



e



2.1.2 Features

In Jimena, networks are usually imported from GraphML files created by the yEd graph editor from yWorks GmbH which is available for free at (yWorks GmbH 2014). The import function as well as all simulation routines support activator-inhibitor-networks as well as full Boolean networks (Figure 3a).

After the network has been loaded, node values and other network parameters such as relative strengths can be set (Figure 3b) and perturbations can be introduced (Figure 3c, 2.1.2.2). The network is then ready to be simulated, and results can be displayed directly in Jimena or exported to the clipboard or a file (Figure 3d).

Many basic network analysis tools such as stable states and the new control centralities are integrated into the graphical user interface (Figure 3e), with many more sophisticated functions such as batch network generation being available in the attached library.

2.1.2.1 Network simulations

In additions to the usual discrete models (ARBN, CRBN, etc., 1.3.3), Jimena supports the continuous models SQDS, BooleCube and (normalized) HillCube (1.3.5 and 1.3.7).

The differential equations in continuous models are solved using a standard forth-order Runge-Kutta method (Burg et al. 2009, pp. 55–63).

2.1.2.2 Perturbations

In Jimena, the value of a node can be forced to equal a given function $f(t)$ independently of the influences from other nodes. Using these perturbations, the reaction to stimulation or repressions of one or more nodes and other aspects of network behavior can be studied.

2.1.2.3 Multithreading

Over the last decade, computers with more than one computational core have become prevalent even in the consumer market.

Multithreaded software is tailored to this environment in that the program defines multiple threads that can run in parallel in the processing cores. All of Jimena's computationally expensive features such as the search for stable states in continuous networks or the calculation of the control centralities have been implemented as multithreaded algorithms (Figure 4, see Appendix 7.2 for technical details).

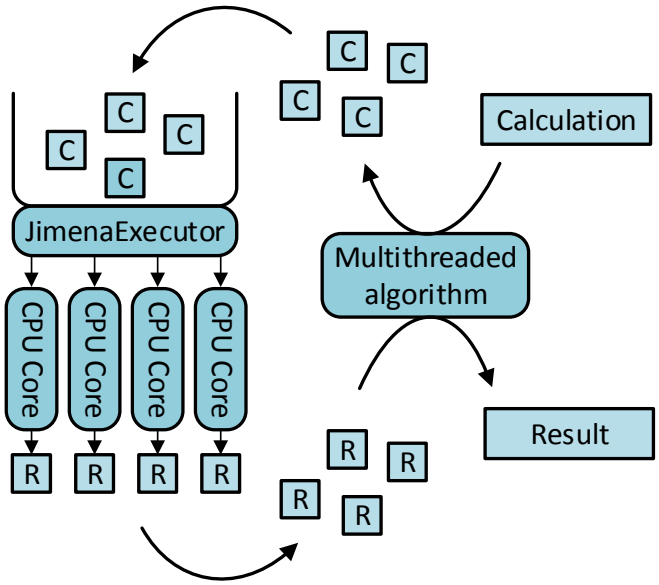


Figure 4: Multithreading in Jimena
Each calculation is divided into a large number of independent subcalculations [C], which are allocated to the computational cores by the JimenaExecutor. Once all subcalculations have finished, the subresults [R] are combined into the result of the whole calculation.

2.1.2.4 Random networks

Jimena's library contains a customizable network generator, which generates random scale-free and Erdős–Rényi network whose number of nodes, number of connections, number of loops and other network parameters can be given by the user (see 3.2 and Appendix 7.1 for technical details).

2.1.2.5 Mutations

Null mutations, also called amorphic mutations (Muller 1932), lead to a complete loss of function of the node or interaction. In Jimena, null mutations of network nodes and connections can be added and removed dynamically, such that the impact of large numbers of mutations can be studied (3.3.6.1).

2.2 Algorithms

2.2.1 The BooleCube interpolation algorithm

In the differential equations of BooleCube and HillCube networks, the BooleCube interpolation polynomial (1.3.7) has to be computed.

To substantially improve on the space and time complexity of this calculation, Jimena uses Boolean trees to represent the Boolean functions of the network.

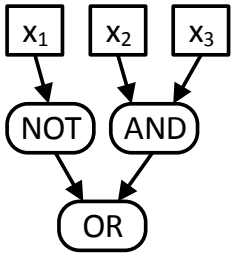


Figure 5: Boolean tree
Boolean tree for the function $B_1(x_1, x_2, x_3)$ = (NOT x_1) OR (x_2 AND x_3). Input variables x_i are connected by Boolean operators. The OR node is the root of the tree, i.e. its value determines the value of the function represented by the tree.

In a Boolean tree (Figure 5), leaves (i.e. nodes without ingoing connections such as x_1 in the figure) are inputs to the function, and non-leaf nodes are unary or binary Boolean gates (such as AND in the figure). Each Boolean gate combines the values from its ingoing connections to an outgoing arrow in accordance with the Boolean function (e.g. AND, OR or NOT) the gate represents. The value of the root node, i.e. the unique node without outgoing connections to other nodes, determines the value of the function.

The first step in a simulation is loading the network into the simulation framework. Boolean trees can be straightforwardly created in linear time complexity $O(E)$ by parsing a Boolean expression (such as "$(\text{NOT } x_1)$ OR $(x_2 \text{ AND } x_3)$"), where E is the length (in characters) of the Boolean expression.

The speed of loading a Boolean function in Jimena therefore depends on its description length E, whereas the theoretical complexity of (at least) $O(2^n)$ exhibited by Odefy (1.6.2) depends on the number of inputs n.

This difference in complexity between Odefy and Jimena is of high practical importance (cf. 3.1.1), since Boolean functions appearing in biological networks can almost exclusively be described by Boolean expressions of moderate length E, presumably due

to limitations in the complexity of biochemical interactions. Most published networks are modeled using simple network design patterns (e.g. activator-inhibitor-patterns), whose functions exhibit expression lengths E in $O(n)$ where n is the arity of the function.

In addition to significant advantages in the creation of the network, the tree structure also expedites the calculation of BooleCube (and therefore HillCube) interpolations by enabling a recursive evaluation of the function.

A recursive algorithm is an algorithm that divides the problem into smaller parts, and then calls itself to compute the solution to these parts. The solutions of the parts are then assembled to a solution of the whole problem.

Jimena's interpolation algorithm recursively evaluates the Boolean tree from the root node to the leaves by applying the interpolation separately to all logic gates.

For a more precise description of the algorithm, consider a regulatory network with nodes $\{x_1,...,x_n\}$. Let the Boolean function $B_k(\ldots)$ of a node x_k be given by a Boolean tree consisting of nodes $\{n_1,...,n_m\}$. Note that as shown in Figure 5, these n_i represent binary or unary Boolean gates, or inputs to the function $B_k(\ldots)$. Each function $B_k(\ldots)$ in the network defines a separate tree and therefore a separate set $\{n_1,...,n_m\}$.

To illustrate the relationship between $\{x_1,...,x_n\}$ and $\{n_1,...,n_m\}$ consider the network $\{x_1,x_2\}$ where $B_1(x_1,x_2) = x_1$ AND x_2 and $B_2(x_1,x_2) = x_1$ OR x_2. A possible Boolean tree for the function B_1 could then be given by the nodes n_1,n_2,n_3, where the root node n_1 is an AND node with the leaves n_2 and n_3, n_2 is an input node representing x_1 and n_3 is an input node representing x_2.

Let the function given by the subtree whose root is n_i be called f_i, such that $f_i(x_j) = x_j$ for some x_j for all input nodes. If a node n_i is not an input node to the network, its binary or unary logic gate is $\otimes_i$. In the example Boolean tree for $B_1(x_1,x_2)$ from above the functions are $f_2(x_1) = x_1$, $f_3(x_2) = x_2$, $f_1(x_1,x_2) = x_1$ AND x_2 and $\otimes_1 = \text{AND}$.

As introduced in section 1.3.7, for an arbitrary Boolean function $B:\{0,1\}^\tau \to \{0,1\}$, $C[B]$ denotes its BooleCube interpolation. A recursive term for the interpolation $C[f_i]$ of a node n_i's function f_i can now be constructed using the following recursive rules:

- If n_i represents an input node of the tree for which $f_i(x_j) = x_j$ the algorithm sets

$$C[f_i] \equiv x_j$$

- If n_i is a unary negating gate (i.e. a NOT) whose input is a node n_j, it sets

$$C[f_i] \equiv 1 - C\left[f_j\right]$$

- If n_i is a binary gate with two inputs n_{j_1} and n_{j_2} (i.e. functions f_{j_1} and f_{j_2}) it sets

$$C[f_i] \equiv \sum_{\overline{a}_1=0}^{1}\sum_{\overline{a}_2=0}^{1}\left((\overline{a}_1 \otimes \overline{a}_2)\cdot\xi\left(C[f_{j_1}],\overline{a}_1\right)\cdot\xi\left(C[f_{j_2}],\overline{a}_2\right)\right)$$

where $\xi(\alpha,\beta) = \alpha\beta + (1-\alpha)(1-\beta)$. Notice that this term collapses to $f_{j_1}\cdot f_{j_2}$ if $\otimes_i = \text{AND}$ and $f_{j_1} + f_{j_2} - f_{j_1}\cdot f_{j_2}$ if $\otimes_i = \text{OR}$, both of which can be calculated very efficiently. The $C[\ldots]$ parts of the terms above are then evaluated using the same rules until all branches of the recursion have reached an input node.

If this algorithm is applied to the root node of the network, the interpolation of the whole Boolean function is returned. See Appendix 7.3 for a proof that the result of this algorithm is identical to the high degree polynomial defined in (Wittmann et al. 2009).

As an example, consider the function B_1 (Figure 5). Traversing the tree starting from the root node n_{OR} yields $C[B_1] = C[f_{OR}] = C[f_{NOT}] + C[f_{AND}] - C[f_{NOT}]C[f_{AND}]$ $= (1-x_1) + x_2x_3 - (1-x_1)x_2x_3\ (= 1 - x_1 + x_1x_2x_3)$.

Determining the time complexity of the algorithm is straightforward, since the interpolation of each node can be computed in a time independent of the number of nodes, i.e. in $O(1)$, and the maximum number of nodes is proportional to the description length, i.e. in $O(E)$. The overall complexity is therefore $O(1)\cdot O(E) = O(E)$. This is a huge improvement over $O\left(2^n\right)$, the minimum time complexity of simulations in Odefy, that directly translates to strongly measurable speed-ups (see section 3.1.1).

2.2.2 Stable states in discrete models

As a side effect, Boolean trees also simplify the creation of binary decision diagrams (BDDs) (Knuth 2011, pp. 202–208) for the Boolean functions of the network.

BDDs represent Boolean functions in directed graphs, similar to Boolean trees (Figure 6). Common problems surrounding Boolean functions, such as finding solutions for a Boolean equation, can be solved efficiently using standard algorithms (Bryant 1986) once a BDD representation has been created, which is not possible with Boolean trees.

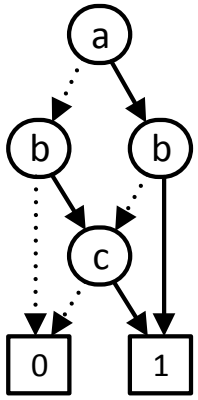


Figure 6: Binary decision diagram (BDD) A BDD for the function (a OR b) AND (b OR c) AND (a OR c). Evaluation starts at the top node "a". If the value of the node is 1, the solid line is followed, if it is 0, the dashed line is followed. For example, for the input values a = c = 1 and b = 0 one would go down from the "a" node to the right "b" node, on to the "c" node and finally along the solid line to the "1" node. This corresponds to (1 OR 0) AND (0 OR 1) AND (1 OR 1) = 1 AND 1 AND 1 = 1.

A possible application of BDDs in regulatory networks is the search for all stable states in discrete models, i.e. network states which reproduce themselves in each following step of a simulation. This approach was previously explored by (Garg et al. 2007) for simple activator-inhibitor networks.

As discussed in section 1.4.1, if B_i are the Boolean functions defining a network consisting of the nodes x_i, a network state $(x_1,...,x_n)$ is a stable state if and only if

$$B_i\left(x_{i,1},...,x_{i,\mathrm{n_i}}\right) = x_i \text{ for all } x_i \quad (*)$$

where $x_{i,j} \in \{x_1,...,x_n\}$ is the j-th input to the function B_i.

To find the stable states of the network, Jimena can therefore determine all network states $(x_1,...,x_n)$ which satisfy the equation (*) by constructing a BDD which represents (*) and then finding all satisfying states using an external BDD library.

In the JavaBDD (Whaley 2007), the library used by Jimena, a BDD for the expression (*) can be constructed from the Boolean trees of the functions B_i in a time proportional to the number of nodes in the Boolean tree. All assignments $(x_1,...,x_n)$ fulfilling the condition (*) can then be computed efficiently by the BDD library.

In essence, while both Boolean trees and BDDs represent Boolean functions and equations, Boolean trees are necessary to speed up the simulation of continuous networks, while BDDs are essential for the efficient calculation of stable states.

2.2.3 Stable states in continuous models

As opposed to discrete networks, stable states in continuous networks cannot be found analytically (Mendoza, Pardo 2010; Wittmann et al. 2009).

It is, however, possible to find a subset of the stable states by simulating the network from random initial states, and collecting the states the network converges to (Mendoza, Pardo 2010). While this algorithm, which is implemented in Jimena, is not guaranteed to return all stable states, the method has the advantage to return an approximation of the sizes of the basins of attraction of the stable states (1.4.2) as a by-product.

Evidently, stable states with very small basins of attraction will not usually be found by the algorithm, but the biological significance of these very "unstable" stable states is doubtful anyhow.

2.3 Analyzing control centrality

This chapter introduces new control centralities tailored to biological networks (1.7).

2.3.1 Convergence and network comparisons

As introduced in section 1.3.3, a time-continuous regulatory network is defined by a vector of nodes, and corresponding differential equations $\dot{x}_i(t) = f_i(x_{i,1},...,x_{i,m})$. Each

function f_i potentially defines multiple arrows in the network graph, i.e. influences of node values $x_{i,1},...,x_{i,m}$ on the node x_i.

In the usual mathematical models, influences can be removed from the network and therefore the network graph, for example by ignoring an activating or inhibiting input when constructing the differential equations of the SQDS method.

To define the control centralities, it is necessary to quantify the difference in behavior of two similar regulatory networks. Comparing time series data of responses to stimuli is not justified due to a ubiquitous lack of kinetic data (Machado et al. 2011; Weinstein,

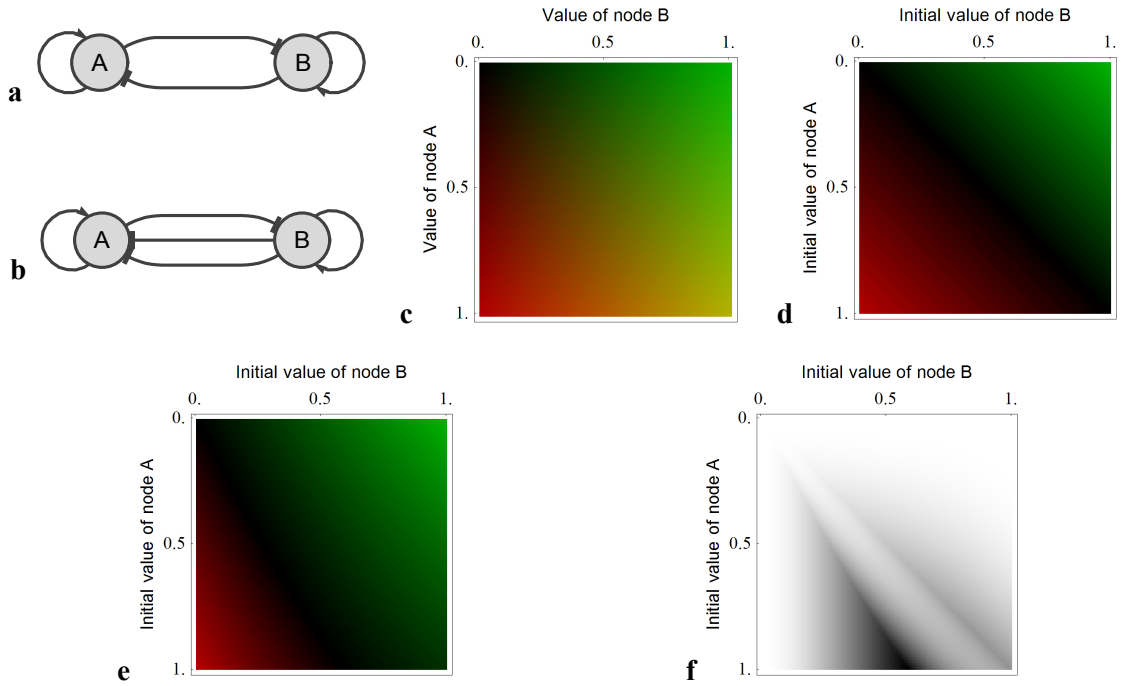


Figure 7: Convergence behavior in regulatory networks
a Artificial regulatory network N_1 with two mutually inhibiting nodes A and B (BooleCube network with default parameters (see 3.2), 10,000 runs per value), a common motif in biological networks. **b** In the network N_2 the strength of the inhibition B → A has been doubled (double arrow). **c** Color scheme encoding the values of nodes A (y-axis) and B (x-axis) into a unique color. **d** Convergence vectors of N_1 in the color scheme c. The network converges to a state where either A or B is activated (lack of yellow in the plot). **e** Convergence vectors of N_2. Node B is noticeably stronger than node A. **f** Mean squared difference of the convergence vectors from 1d and 1e in a linear color scheme (0 is white, 1 is black). Changes in network behavior between the two networks are most prominent (i.e. black), if node A is fully activated and node B's activity is intermediate.

Mendoza 2012). The new definitions are therefore based on the convergence behavior ψ of the network, which exhibits robustness against kinetic modeling uncertainties.

If $\boldsymbol{x} \in [0,1]^n$ is a state vector of a network N, let $\psi_N(\boldsymbol{x})$ be the state vector the network N converges to when simulated for time indices $t \to \infty$ from the initial state $\boldsymbol{x}$ (cf. 1.4). The codomain of ψ_N are the stable states of the network, since the function is undefined if the network state does not converge.

To quantify the difference between two (state) vectors $\boldsymbol{x}, \boldsymbol{y} \in [0,1]^n$, the mean squared difference $\mu_S(\boldsymbol{x}, \boldsymbol{y}) = |S|^{-1} \cdot \sum_{x_i \in S} (x_i - y_i)^2$ is used, where $S \subseteq \{x_1, ..., x_n\}$ represents a subset of the nodes which are considered significant, for example biologically relevant effector nodes. If no significant nodes are given, all nodes are considered significant.

The convergence difference of two networks N_1 and N_2 with identical nodes but potentially different topologies is then defined as

$$\Delta_S(N_1, N_2) = \int_{\boldsymbol{x} \in [0,1]^n} \mu_S(\psi_{N_1}(\boldsymbol{x}), \psi_{N_2}(\boldsymbol{x}))d\boldsymbol{x}$$

Put simply, the equation compares the convergence behavior of the two networks for all possible initial states. It is implicitly assumed that the integration domain in all integrals is limited to points where the integrand exists (i.e. the network converges), and that the results of the integrations in these cases are scaled such that they represent the mean value of the integrand. In practice, Jimena approximates the integrals numerically by random sampling. Note that the existence of this and other integrals in this chapter cannot be guaranteed in a strict mathematical sense. The approximations do, however, converge in practice.

For example, in a gene expression network, the convergence function ψ corresponds to letting the system settle to a steady state, where no more changes in gene expressions levels occur without new external stimuli. N_1 and N_2 could be two genetic regulatory networks, where N_1 is the wild-type system, and N_2 models the nonsense mutation of a gene. The convergence difference Δ then quantifies how much the steady states of the two dynamic systems differ, if both systems are started from the same initial state $\boldsymbol{x}$. If

substantial differences occur for many initial states, the mutation affects the behavior of the regulatory network.

The concept is explained in Figure 7. The convergence functions $\psi(\boldsymbol{x})$ of two small artificial networks (Figure 7a,b) are plotted in Figure 7d and e using the color scheme from Figure 7c. The mean squared difference $\mu\left(\psi_{N_1}(\boldsymbol{x}), \psi_{N_2}(\boldsymbol{x})\right)$ of the two networks for initial vectors $\boldsymbol{x} \in [0,1]^2$ is given in Figure 7f, and integration over this plot yields the convergence difference $\Delta(N_1, N_2)$.

The convergence difference can also be used as a general measure for the similarity of two networks, extending previous approaches to network comparisons based on network topology (Morris et al. 2014) by taking actual network response behavior into account.

2.3.2 Calculating total centrality

Using the convergence difference Δ_S, the influence of network manipulations on the behavior of a network can be quantified. If N is a regulatory network, and $N_{del(x_a \to x_b)}$ is identical to N apart from the deletion of the connection from node x_a to node x_b, the total (control) centrality (TC) $TC_{N,S}(x_a \to x_b) = \Delta_S\left(N, N_{del(x_a \to x_b)}\right)$ is a measure of the influence of the connection on convergence behavior. After deleting all connections originating in a node x_a resulting in the network $N_{del(x_a)}$, the total centrality of the node is $TC_{N,S}(x_a) = \Delta_S\left(N, N_{del(x_a)}\right)$, which quantifies the impact of null mutations of x_a.

The concept is illustrated in Figure 8.

The vulnerability V of a network, a measure of the susceptibility of the network to mutations, may be intuitively defined as the mean of the total centralities $n^{-1} \cdot \sum_{x_a} TC_N(x_a)$. Conversely, $1 - V$ is a measure of the robustness of the network.

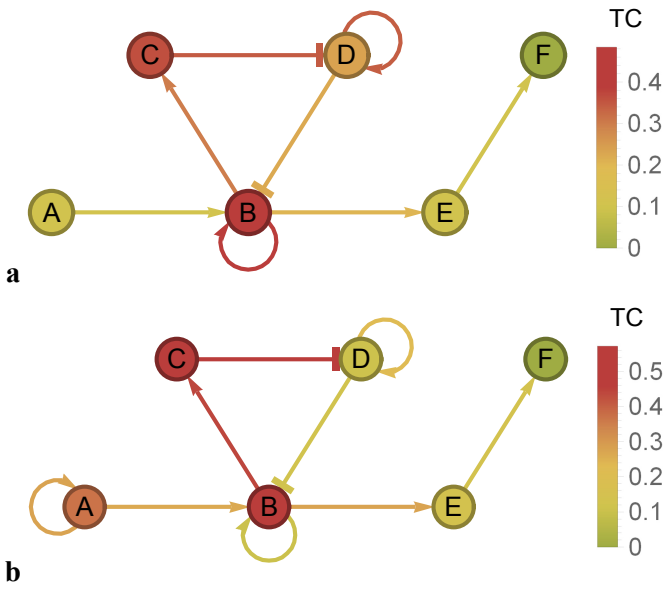


Figure 8: Total centrality in an artificial network
a Total control centrality of nodes and connections in an artificial network (normalized HillCube network with default parameters (see 3.2), 10,000 runs per value). The influence of node A is limited (TC_A = 0.11) since its value converges to 0 due to the decay component in the Odefy model (1.2.5). Mutations of node B, on the other hand, strongly affect network dynamics (TC_B = 0.48) by disrupting the central cycle B → C → D as well as disrupting the A → B → E pathway. Node E has a far smaller influence (TC_E = 0.10) since it is only involved in the B → E → F output pathway. **b** Adding a loop to node A enables it to hold a state, greatly increasing its influence (TC_A = 0.37) and the influence of the downstream nodes B (TC_B = 0.57) and E (TC_E = 0.14). The influence of node D decreases with a strong node A (TC_D = 0.11, 0.24 without the additional loop).

2.3.3 Calculating value centrality

An analogous concept which quantifies the influence of the value of a node x_a, which is assumed to be the last node in the state vector for the sake of simplicity, is the value (control) centrality (VC) defined by

$$VC_{N,S}(x_a) = \int_{\boldsymbol{x}\in[0,1]^{n-1}} \int_{a\in[0,1]} \int_{b\in[0,1]} \mu_S(\psi_N(\boldsymbol{x},a),\psi_N(\boldsymbol{x},b))\,db\,da\,d\boldsymbol{x}$$

In short, value centrality examines whether the convergence of the network changes if the activation of one node is manipulated. From a biological point of view, the formula

corresponds to artificially altering the gene expression level of a node x_a in a gene expression network, and examining whether the network state is affected in the long term.

To define the value centrality of single connections, the node the connection originates in has to be split. For a connection $x_a \to x_b$ in a network N with nodes $x_1,...,x_n$, the network $\Upsilon_N(x_a \to x_b)$ with nodes $x_1,...,x_n,x_{n+1}$ and functions $\tilde{f}_i$ is constructed, where $\forall i \notin \{n+1,b\} : \tilde{f}_i(\boldsymbol{x}) = f_i(\boldsymbol{x})$, $\tilde{f}_{n+1}(\boldsymbol{x})$ is equal to $f_a(\boldsymbol{x})$ except for all occurrences x_a which are replaced by x_{n+1} and $\tilde{f}_b(\boldsymbol{x})$ is equal to $f_b(\boldsymbol{x})$ except for all occurrences of x_a which are replaced by x_{n+1}. The value centrality of the connection is then given by

$$VC_{N,S}(x_a \to x_b) = \int_{\boldsymbol{x}\in[0,1]^n} \int_{a\in[0,1]} \int_{b\in[0,1]} \mu_S\left(\psi_{\Upsilon_N(x_a\to x_b)}(\boldsymbol{x},a), \psi_{\Upsilon_N(x_a\to x_b)}(\boldsymbol{x},b)\right) db\, da\, d\boldsymbol{x}$$

In simpler terms, the node x_a is split into two identical nodes x_a and x_{n+1}, where x_{n+1} takes the place of x_a only for the connection $x_a \to x_b$, after which the influence of the value of x_{n+1} on the convergence behavior is quantified.

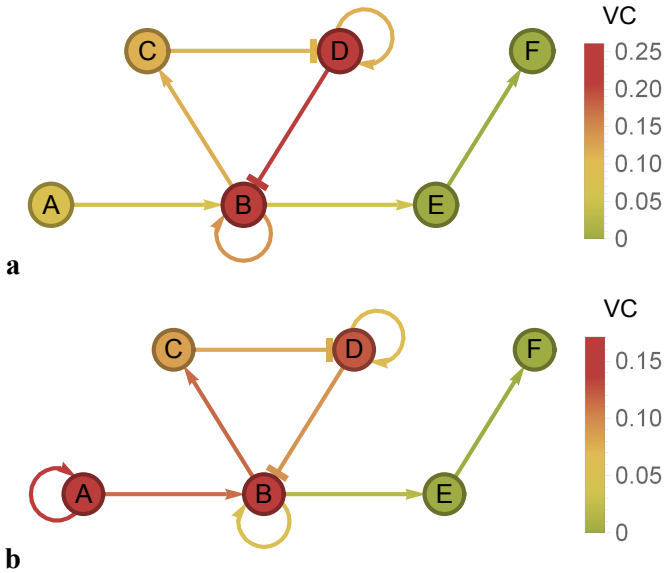


Figure 9: Value centrality in an artificial network
a The value of node B strongly influences the convergence of the network ($VC_B = 0.21$) since B can hold its state due to the self-amplifying loop and distribute it via the connections B → C and B → E. The value of node E on the other hand is overridden by the B → E connection ($VC_E < 10^{-10}$). **b** Adding a loop to node A increases the influence of its value at the expense of other nodes ($TC_A = 0.17$, 0.08 without the additional loop).

See Figure 9 for an example.

The mean of the value centralities $n^{-1} \cdot \sum_{x_a} VC_N(x_a)$ is a measure of how well the network can be steered by the value of its nodes, i.e. its controllability.

2.3.4 Calculating dynamic centrality

Deleting a connection from a network intuitively removes two types of influence from the network, the influence of the value of the node (i.e. value centrality) and a dynamic influence caused by its effect on network dynamics. The latter concept is especially difficult to define since in some network models such as SQDS, a connection $x_a \to x_b$ exerts a constant influence even beyond the value of its source node, i.e. there is no possible value $y \in [0,1]$ of the source node x_a which would mimic the effect of completely deleting the connection.

If specific values for x_b and its input nodes are assumed, such a neutral value for x_a can, however, usually be found for this specific situation. It is therefore possible to define the dynamic centrality $DC_{N,S}(x_a \to x_b)$ of a connection $x_a \to x_b$ by constructing $\Upsilon_N(x_a \to x_b)$ (cf. 0) and determining

$$\int_{\boldsymbol{x} \in [0,1]^n} \min \left\{ \mu_S \left(\psi_{\Upsilon_N(x_a \to x_b)_{del(x_{n+1} \to x_b)}}(\boldsymbol{x}, y), \psi_{\Upsilon_N(x_a \to x_b)}(\boldsymbol{x}, y) \right) \middle| y \in [0,1] \right\} d\boldsymbol{x}$$

In other words, the convergence behavior between the network, where the connection has been deleted, and the network where its source x_{n+1} has been initialized with the optimal neutral value y, is compared. If the difference is high, the connection influences the network independently of its initial value, for example by relaying the stimuli of other nodes.

This definition is more difficult to grasp from a biological point of view. As an example, consider the regulation of blood glucose levels in humans by insulin and other hormones such as glucagon or incretins. For a given situation of this dynamic system, the definition tries to adjust the initial insulin concentration such that the system behaves as if insulin was not involved in the regulation of blood glucose at all. If this succeeds, only the initial insulin concentration affects the systems. If it fails, which is likely in this scenario, insulin affects the system beyond its initial concentration, for example since it relays regulatory stimuli from other parts of the system. This property is called dynamic centrality.

An analogous measure $DC_{N,S}(x_a)$ for nodes is defined by splitting the node x_a with k outgoing connections into k nodes $x_{a,1},...,x_{a,k}$ by repeated application of $\Upsilon_N(x_a \to ...)$ and searching for optimal neutral initial values $y_1,...,y_k \in [0,1]$ for all those connections.

See Figure 10 for an example.

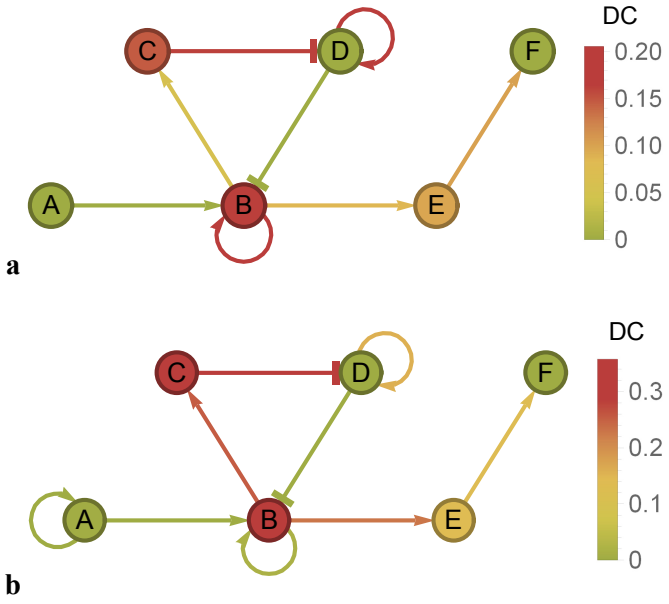


Figure 10: Dynamic centrality in an artificial network
a Dynamic centrality in the network from Figure 8 is held by nodes with essential relaying functions such as nodes B ($DC_B = 0.16$) or E ($DC_E = 0.10$). Intriguingly, the dynamic influence of node D is very low ($DC_D < 10^{-10}$). This is plausible, however, since due to the double negation in the B → C → D → B loop, node B will only be influenced in the direction of the value it already holds. The D → D loop on the other hand is relevant for the value of node D itself. Note that dynamic centrality deals with relaying functions from other nodes only, which is why the A → B connection is dynamically irrelevant ($DC_{A\to B} < 10^{-10}$) as there are no nodes upstream from A. **b** Network dynamics are not changed significantly by adding a loop to node A.

2.3.5 Pairwise influences and sensitivities

Restricting the calculation of the network difference to a node x_c, results in the total centrality, value centrality and dynamic centrality of a connection $x_a \to x_b$ or a node x_a on the node x_c:

$$\Theta_N(x_a \to x_b, x_c) = \Theta_{N,\{x_c\}}(x_a \to x_b)$$
$$\Theta_N(x_a, x_c) = \Theta_{N,\{x_c\}}(x_a)$$
$$\text{where } \Theta \in \{TC, VC, DC\}$$

The sensitivity of a node x_c with respect to a certain centrality Θ is then defined as $S_{N,\Theta}(x_c) = n^{-1} \cdot \sum_{x_a} \Theta_N(x_a, x_c)$. The value sensitivity $S_{N,VC}$ quantifies how much the value of a node varies if manipulations of the initial vector change the convergence behavior, and is therefore high for nodes whose values separate different stable states and are a good representation of the current network status. $S_{N,TC}$ on the other hand is a measure of the susceptibility of the node to mutations in the network (Figure 11).

2.3.6 Interpretation of the centrality metrics

The impact of mutations leading to permanent overexpression, underexpression or functional defects of network nodes is best represented by their total centrality, whereas value centrality captures the influence of the value of the node in the unmutated regulatory system.

Total centrality thus covers long acting pharmacological agents as well as germinal and somatic mutations affecting promoter activity, protein function, and so on. Short acting regulatory effects like transient receptor activity or temporary upregulation of a transcription factor are best represented by value centrality.

The more abstract dynamic centrality, on the other hand, usually constitutes a large part of the total centrality of powerful nodes, representing their strong effects on network dynamics. In signaling cascades, where relaying and integrating functions outweigh other node roles, dynamic centrality is especially prevalent.

Since the absolute values of the centralities depend on the overall sensitivity of the network, values are interpreted best relative to each other. As a rough guide, in medium sized networks (20 to 1000 nodes) centralities above 10^{-6} represent strong influences, centralities between 10^{-6} and 10^{-10} are intermediate, and centralities below 10^{-10} are weak.

If the value centralities of one or more nodes are considerably stronger than other value centralities in the network, these nodes can be said to dominate the behavior of the network, as the influence of the remaining nodes is negligible in comparison. In a biological context, dominating nodes often represent essential master transcriptional regulators (see 3.3.3).

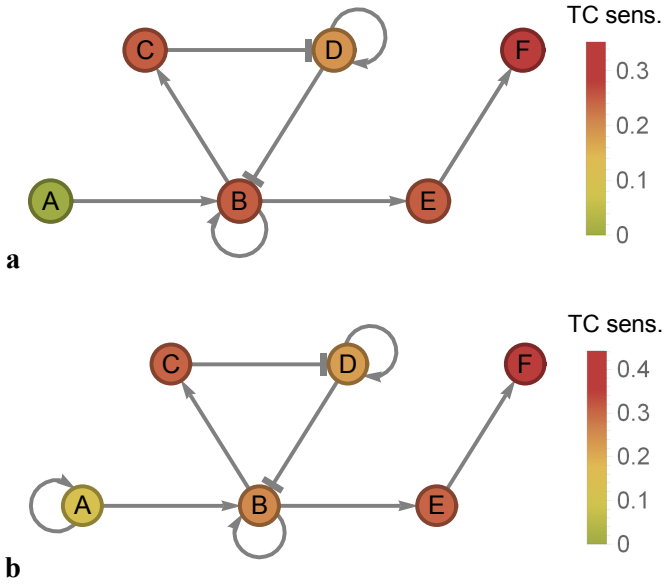


Figure 11: Total centrality sensitivity in an artificial network
a The sensitivity to mutations $S_{N,TC}$ is higher for nodes more downstream, with the most sensitive node being node F. **b** Adding a loop to node A enables it to hold a state, which renders the node (moderately) sensitive to mutations (of the A → A loop).

2.4 Extended concepts

2.4.1 Dynamically connected networks

Based on dynamic and total centrality, dynamically connected networks (DCNs) are a tool to understand the subnetwork interconnecting a subset of the nodes. They are defined as networks $\tilde{N}$ containing a subset of the connections (and therefore the nodes) of a given network N which mimics the convergence behavior of N with respect to a set of significant nodes $S \subseteq \{x_1,...,x_n\}$. This definition is explained further in Figure 12.

For a formal definition, consider the function

$$\xi_{N,S}(x_a \to x_b) = \begin{cases} TC_{N,S}(x_a \to x_b) & if\ x_a \in S \\ DC_{N,S}(x_a \to x_b) & else \end{cases}$$

which returns the total centrality of connections originating in a significant nodes, and the dynamic centrality of other nodes. A DCN algorithm now successively deletes connections $x_a \to x_b$ from the network N for which $\xi_{N,S}(x_a \to x_b) \leq \Lambda$ where $\Lambda \approx 0$ is a low threshold which depends on the computational accuracy of the calculations. All

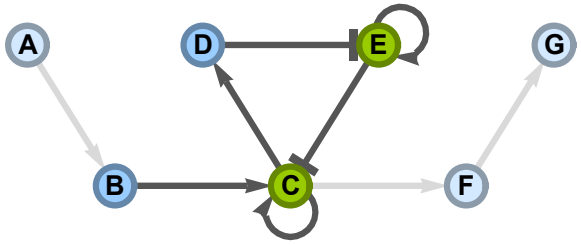


Figure 12: Minimal dynamically connected networks
An artificial network (SQDS method with default parameters) is reduced to a minimal dynamically connected network (darker gray, $\Lambda = 10^{-10}$) if the nodes C and E are significant. The interactions C → F and F → G are removed since they lead to the non-significant output nodes F and G, and are therefore irrelevant for the convergence of C and E. A → B is not dynamically relevant for the convergence either. Interestingly, the connection B → C is preserved since in the SQDS method it influences the differential equations of node C even beyond the value of B. In other words, only the nodes B, C, D and E are necessary for all interactions between the nodes C and E.

of these steps yield valid DCNs. If no more deletions are possible, the remaining network $\tilde{N}$ is a minimal DCN (MDCN) of the network N with respect to the set of significant nodes S, i.e. a minimal subnetwork of N which is sufficient to model all interactions between the significant nodes.

2.4.2 Directional value centrality and cooperativity

As an extension of value centrality, the direction of the shift in network behavior as a function of the direction of the manipulation of the initial values can be quantified.

The first step is to extend the definition of the mean squared distance to yield a component-wise result and to include the direction of the influence:

$$\boldsymbol{\mu}^*(\boldsymbol{x},\boldsymbol{y}) = (\boldsymbol{y}-\boldsymbol{x}) \circ |\boldsymbol{y}-\boldsymbol{x}|$$

where "$\circ$" is the component-wise vector product (Hadamard product) for which $(a,b)\circ(c,d) = (a\cdot c, b\cdot d)$. The directional value centrality of a node x_a which is the last node in the state vector then is

$$\boldsymbol{VC}_N^*(x_a) = \int_{\boldsymbol{x}\in[0,1]^{n-1}} \int_{a\in[0,1]} \int_{b\in[0,1]} (b-a)\cdot \boldsymbol{\mu}^*(\psi_N(\boldsymbol{x},a),\psi_N(\boldsymbol{x},b))\, db\, da\, d\boldsymbol{x}$$

providing information on how increases or decreases in the value of x_a affect the convergence of all nodes in the network.

As an example, consider the network from Figure 13 with nodes (A, B, C). The directional value centralities in the network (40,000 runs per centrality) are

$$\boldsymbol{VC}_N^*(A) = (0.19, 0.00, 0.08)$$
$$\boldsymbol{VC}_N^*(B) = (0.00, 0.19, -0.16)$$
$$\boldsymbol{VC}_N^*(C) = (0.00, 0.00, 0.00)$$

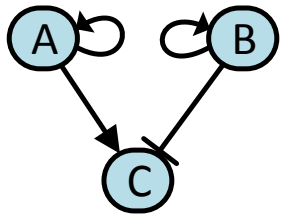


Figure 13: Directional value centrality
a Small artificial network (SQDS network with default parameters, see 3.2) to demonstrate directional value centrality.

The node A therefore wields activating influence on itself ($\boldsymbol{VC}_N^*(A,A)=0.19$) and on node C (0.08) whereas node B wields activating influence on itself (0.19) and inhibiting influence on node C (-0.16). C does not influence any other nodes.

This definition can be extended to connections using the node split function Υ_N analogously to 0.

Using directional value centrality, the functions of two network nodes can now be compared by applying Pearson's population correlation coefficient $r(\boldsymbol{x},\boldsymbol{y})$ (Eid et al. 2011, p. 506) to the $\boldsymbol{VC}_N^*$ vectors of two nodes, or to a significant subset of the components of $\boldsymbol{VC}_N^*$. Due to the characteristics of Pearson's correlation, this metric is above 0 for nodes with synergistic effects (positive cooperativity), and below 0 for nodes with antagonistic effects (negative cooperativity). For example, in the network from Figure 13, nodes A and B are strong antagonists with $r\left(\boldsymbol{VC}_N^*(A),\boldsymbol{VC}_N^*(B)\right)=-0.45$.

2.4.3 Transforming centrality values

For a researcher, it is easy to grasp that a centrality of 10^{-1} represents a strong influence, 10^{-6} represents a medium influence (see 2.3.6), and that centralities of 10^{-16} and 10^{-18} are both irrelevant since they are below the accuracy of about 16 significant figures in the double-precision floating-point calculations used by Jimena (IEEE 2008).

For some plots, it is expedient to use this notion to normalize non-zero (directional) centrality values in the domain $(-\infty,\infty)$ to the codomain $(-1,1)$ by the function

$$\eta(x)=\frac{x}{|x|(1-log_{10}|x|)}$$

For example, for the values above, the transformation yields $\eta\left(10^{-1}\right)=0.5$, $\eta\left(-10^{-1}\right)=-0.5$, $\eta\left(10^{-6}\right)=0.143$, $\eta\left(10^{-16}\right)=0.059$ and, $\eta\left(10^{-18}\right)=0.053$.

By component-wise application, $\eta(x)$ can be extended to vectors instead of scalar values.

3 Results

In this chapter, Jimena and the control centralities are applied to artificial and biological networks. For the sake of intelligibility, methods specific to each example and a short discussions of the results, as well as links to experimental results, are included here.

3.1 Speed of the Jimena framework

3.1.1 Network simulations

In section 2.2.1, a strong acceleration of BooleCube interpolations by Jimena's tree interpolation algorithm was deduced theoretically. To measure this speed-up in different simulations frameworks, a scalable network topology is used here, i.e. a network definition that enables the creation of similar networks of different sizes.

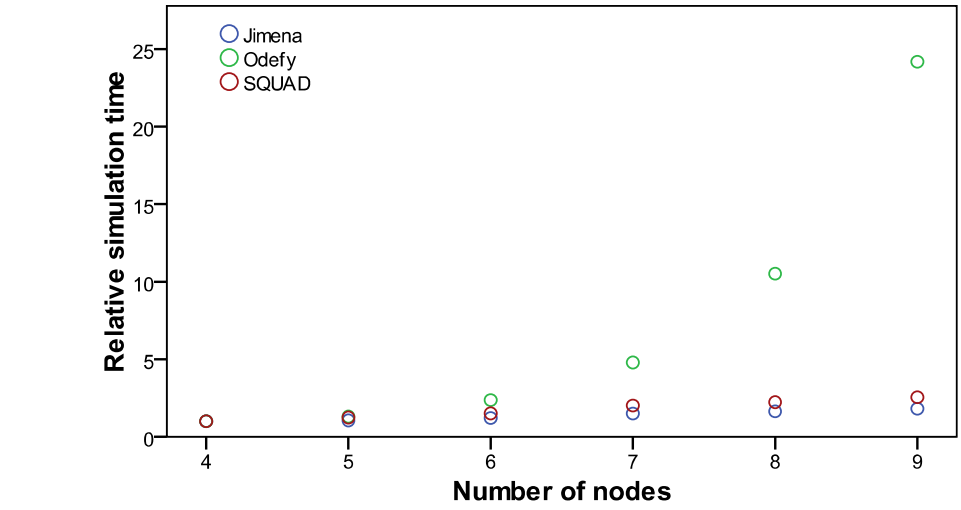
Specifically, the frameworks were compared using artificial networks with an even number n of nodes $x_1,...,x_n$ whose Boolean functions are

$$B_i = \begin{cases} x_3 \text{ OR } ... \text{ OR } x_n & : i \le 2 \\ x_i & : 3 \le i \le n/2+1 \\ 0 & : n/2+1 < i \end{cases}$$

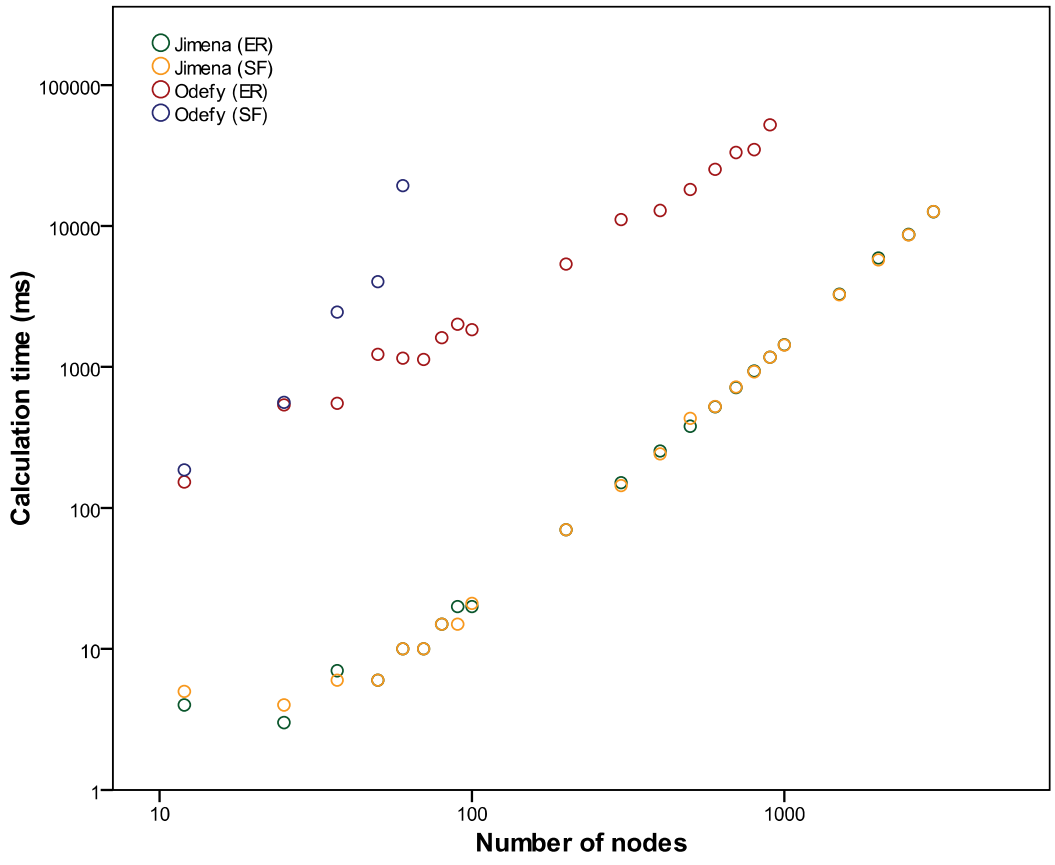
This simple topology results in a maximum node (in-)degree of $n-2$, $2.5n-5$ interactions and $2^{(n-2)/2}$ stable states.

The networks were simulated in Jimena and Odefy (Figure 14a) for 10 simulation time units as normalized HillCube networks. Since the OR operator in the Boolean functions can be seen as an integration of activating inputs, the network can also be interpreted as an activator-inhibitor network and SQUAD could be included in the comparison.

All simulation time benchmarks have been performed on a Windows 7 system running Java SE 7 with a 4-core 2.67 GHz Core 2 Quad Q9400 CPU (energy saving states deactivated) and 4 GB of RAM.



a



b

Figure 14: Simulation speed in continuous models
a Time to simulate a standardized network (cf. text) with the given number of nodes (x-axis). Note that in all figures the number of nodes refers to the number of actual network nodes as opposed to the number of nodes in the Boolean tree. The data series are scaled to coincide for a network with 4 nodes. Actual simulation times for 4 nodes: Jimena = 0.019 s, Odefy = 0.040 s, SQUAD = 0.046 s. **b** Random BooleCube Erdős–Rényi and scale-free networks with a given number of nodes n and 3·n interactions were simulated for 10 simulation time units with a step size of 0.05 s in Jimena. All simulations were aborted after a calculation time of 1 hour.

Directly comparing the simulation speeds of Odefy and Jimena is not trivial, since Odefy's computational accuracy decreases when simulating longer simulation times t, which is not the case in Jimena.

To avoid such biases, the values in Figure 14a were scaled linearly such that the data series coincide for the network with $n = 4$ nodes. Evidently, Jimena's time complexity reacts much more benignly than Odefy's to high node degrees, and even better than SQUAD which only simulates an activator-inhibitor network instead of a Boolean one.

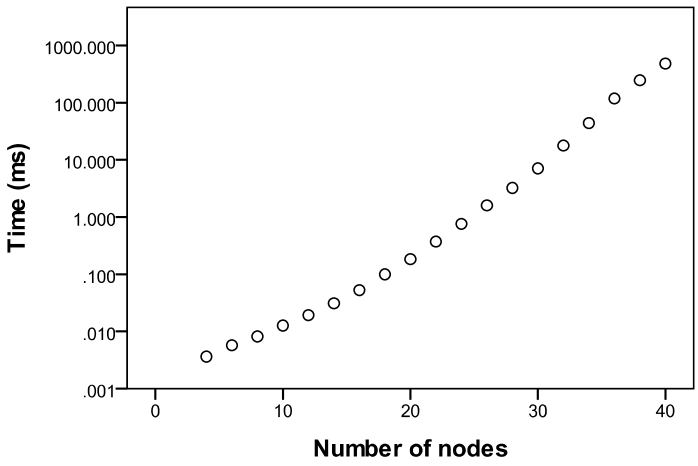


Figure 15: Stable state calculation speed in Jimena
Time needed to determine the discrete stable states of a scalable network topology (cf. 3.1.1) with $2^{(n-2)/2}$ stable states per node. Note that for n = 40 the maximum node degree is 38, i.e. very high.

While this example shows Jimena's performance for high node degrees, it does not cover networks with large numbers of nodes. A second benchmark (Figure 14b) therefore compares the runtime of BooleCube interpolations in Odefy and Jimena in small to large size random Erdős–Rényi and scale-free networks (see section 1.5). Note how Odefy performs especially unfavorably for scale-free networks, which by definition tend to exhibit high node degrees, while Jimena's runtime again reacts benignly to increases in node degree, number of connections and number of nodes. The simulation of scale-free networks with 70 nodes or more did not terminate in Odefy even after runtimes of several hours. Similar experiments with large HillCube and normalized HillCube networks yielded almost identical results.

3.1.2 Stable states in discrete networks

Since the number of stable states in discrete models can grow exponentially with the number of nodes, the theoretical runtime of a stable state search algorithm is also exponential. To benchmark Jimena's ability to find stable states in discrete models, the scalable network topology from section 3.1.1 was used. It features $2^{(n-2)/2}$ stable states for n nodes which is not much lower the theoretical maximum of 2^n.

Figure 15 shows that the time needed to calculate stable state stays very low even for high node degrees. It grows linearly with the number of results and reaches noticeable runtimes only for very high numbers of stable states (40 nodes: 524,288 stable states). Note that the exponential increase of the number of stable states with the size of the network is an artificial worst case scenario, and that for evolutionary reasons many regulatory systems only feature a limited number of stable states.

Further experimentation showed that for medium sized random scale-free networks (100 nodes, 200 interactions, 100 unique networks) Jimena achieves a mean run time of 3890 ms (median 1401 ms), and that the calculation of stable states in discrete networks in Jimena is usually possible for random networks until about 150-200 network nodes and 500 interactions.

Since networks for which Jimena takes a measurable time to calculate the stable states cannot be loaded in Odefy, the two frameworks could not be compared in this respect.

3.2 Control centrality in random networks

Due to the limited number of available regulatory networks, general principles of control can be investigated best in large numbers of random networks. In this chapter, control centrality in random Erdős–Rényi and scale-free (see 1.5) networks is examined.

In Jimena's network generator, Erdős-Rényi networks are created by randomly distributing the desired number of edges over the network nodes (Erdős, Rényi 1959) and discarding networks which are not fully connected.

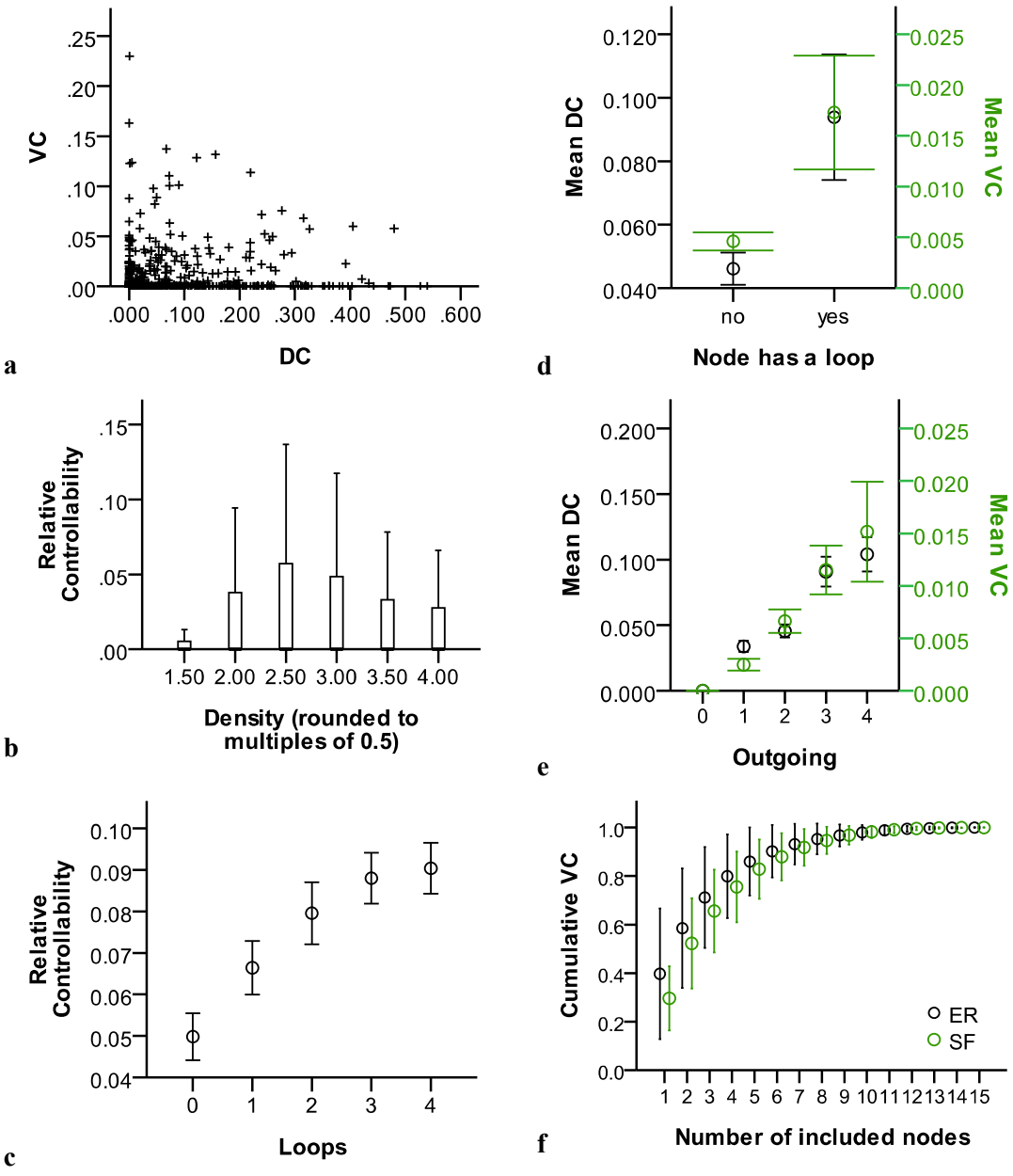


Figure 16: Controllability and centrality in random networks
a There is almost no correlation between dynamic centrality (DC) and value centrality (VC) in random SF and ER networks (15 nodes, 25 connections). **b** High controllability-vulnerability-ratios (relative controllabilities) in random SF networks (20 or 40 nodes, 30-80 or 60-160 connection, respectively) are best achieved in sparsely connected networks with densities around 2.5 (box plot according to the SPSS definition, upper end of the box is the 75% percentile, densities rounded to multiples of 0.5, outliers have been removed) **c** Self-amplifying loops increase relative controllability in SF networks (20 nodes, 50 connections, error bars ±1SE). **d** Loops increase dynamic centrality and value centrality of nodes in random SF and ER networks (15 nodes, 25 connections, error bars ±1SE). **e** Outgoing connections increase value and dynamic centrality in random SF and ER networks. (15 nodes, 25 connections, error bars ±1SE) **f** Cumulative value centrality of the most influential nodes in random SF and ER networks (15 nodes, 25 connections, 0 to 4 loops). The influence resides in only a small number of nodes (error bars ±1SD).

Scale-free networks were grown from a small seed network to the desired size by an algorithm called preferential attachment, in which the probability that a node receives a new connection is proportional to the number of connections it already has (Barabási, Albert 1999) (see Appendix 7.1 for more details).

Unless otherwise specified, network parameters were set to a decay constant of 1 in the SQDS and the Odefy models, parameters n = 2 and k = 0.5 for Odefy's Hill function, and a steepness parameter of h = 10 in the SQDS model for all network in Chapter 3.

The association between dynamic centrality and value centrality is significant ($p < 0.001$, Pearson's correlation, n = 1324) but very weak ($\overline{R}^2 \approx 0.015$, linear regression, Figure 16a). This intriguing observation shows that although both centralities measure the network influence of a node, they capture almost independent aspects of its function. In a linear regression analysis, dynamic and value centrality account for $\overline{R}^2 \approx 0.98$ of the variance in the total centrality, which confirms that they are its essential components. The centralities are thus tightly linked, and high controllability (2.3.3) also necessitates a certain amount of vulnerability (2.3.2).

To assess the balance between a controllable and a robust network topology, it is useful to calculate the relative controllability of the network, defined as the ratio between its controllability and its vulnerability. Considering this ratio helps to control for distorting effects of (functionally) disconnected nodes which appear as artefacts in automatically assembled networks and artificially decrease both their vulnerability and controllability.

In random networks, the relative controllability of scale-free networks is maximal for networks where the density (connections to nodes ratio) is between 2.0 and 3.0 (Figure 16b). According to the data, controllability also has a pronounced maximum in this range, which suggests that a network is controlled better if sparsely connected.

Densities between 2.0 and 3.0 are prevalent in biological networks, with a review (Leclerc 2008) reporting values between 1.4 and 2.75 and a survey in section 3.3.7 finding densities between 1.3 and 3.4. High relative controllabilities are biologically plausible since networks with low controllability would be too stiff to react to internal and external

stimuli thus impairing their regulatory function, while high vulnerability would render the system more susceptible to deleterious mutations. It is therefore not surprising that the relative controllability of biological networks (3.3.7) is high compared to random networks included in Figure 16b.

Figure 16c shows that self-amplifying loops (1.2.4) greatly increase relative controllability ($p < 0.001$). This may contribute to their high conservation in evolutionary processes (Kiełbasa et al. 2008). The influence is mainly exerted by increasing the mean value centrality ($p = 0.006$) and not by changing the mean total centrality ($p = 0.24$, all Pearson's correlation, $n = 1{,}199$). This association can also be traced back to single nodes (Figure 16d), where nodes with a self-amplifying loop have higher dynamic centrality ($p = 0.002$), value centrality ($p < 0.001$) and, interestingly, total centrality ($p < 0.001$, all independent samples t-tests, $n = 1{,}324$).

Another interesting measure is the Gini coefficient (Dixon et al. 1987; Dixon et al. 1988)

$$Gini(x_1,...,x_n) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}|x_i - x_j|}{2n^2\sum_{i=1}^{n}x_i} \text{ (for values } x_1,...,x_n)$$

of the distribution of the centralities, a common measure for inequality, which is 0 if the influence is evenly distributed and approaches 1 for increasingly unequal distributions.

In scale-free networks, a mean Gini coefficient of 0.71 is obtained for the total centralities and 0.68 for the value centralities, indicating very uneven distributions where the centrality is held by just a handful of nodes. For example, an average of 2 out of 15 nodes in random Erdős–Rényi and scale-free networks contain more than half of the total value centrality (Figure 16f). These results are confirmed by biological networks (3.3.7).

The properties of dominating nodes (2.3.6) are controversial, with (Liu et al. 2011) asserting that high degree nodes in linear networks are rarely driver nodes. To shed light on this issue, it is reasonable to distinguish the number of connections originating in a node from the number of connections leading to it. Outgoing connections (Figure 16e) substantially increase the value centrality ($p < 0.001$) and the dynamic centrality

($p < 0.001$), while incoming connections only increase the dynamic centrality ($p < 0.001$) but not the value centrality ($p = 0.09$, all Pearson's correlation, $n = 1{,}324$), which is plausible since value centrality does not include relaying functions.

These results strongly suggest that on average high degree nodes influence network behavior more than low degree nodes, not supporting (Liu et al. 2011). This is plausible in biological systems, where the strength of a transcription factor increases with the number of target genes, as well as in artificial models, where the strength of a node increases with its appearances in the differential equations of other nodes.

3.2.1 Influence of network parameters

In mathematical models such as HillCube or SQDS (1.3), the network is not only determined by its topology, but also by numerical values describing the interactions such as the relative influence weights ω_i in SQDS networks.

In as shown in Figure 17, total centralities in random networks react benignly even if all influence weights are changed. The relative change is proportional to the relative change in the parameters. Considering that strong nodes have centralities in the order of 10^{-1}, whereas weak nodes exhibit values smaller than 10^{-8}, relative changes of ±50% induced by parameters changes are negligible.

Dominating nodes and other network functions are thus encoded in the topology and not in the parameters, providing robustness against modeling uncertainties regarding kinetic parameters.

3.3 Analysis of control centrality in biological networks

This chapter discusses control centrality in regulatory networks from different biological kingdoms. As needed, each analysis is prefaced with a short description of its biological background, with functional and biochemical details being available in the references.

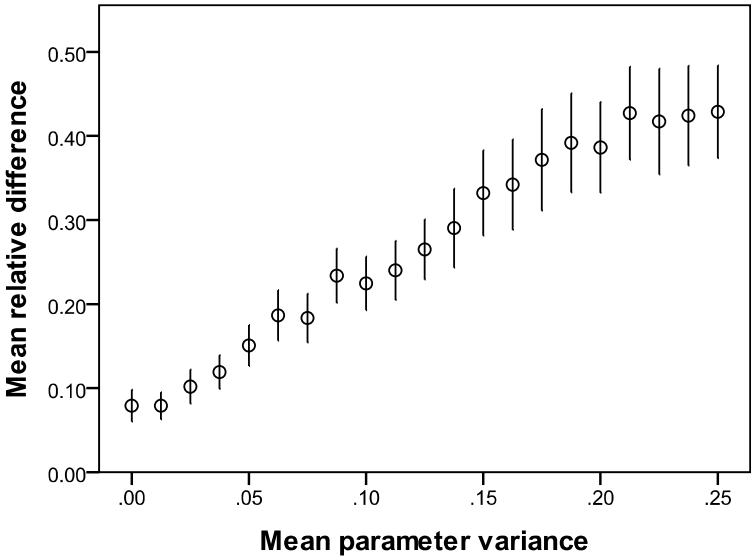


Figure 17: Influence of network parameters
In random, scale-free SQDS networks N_{orig} with 8 nodes and 15 connections (excluding input loops), the total centrality of all nodes was determined. The relative weight of all influences in the network was then multiplied by a random value in the range $1 \pm 2 \cdot$Mean parameters variance resulting in the altered network N_{var}, and all total centralities were recalculated. Plotted is the mean of all relative differences $2 \cdot |TC_{Norig}(x_i)/TC_{Nvar}(x_i)| / (TC_{Norig}(x_i)+TC_{Nvar}(x_i))$ between the total centralities in the two networks as a function of the mean parameter variance (50 networks per mean parameter variance, 1000 runs per total centrality, error bars ±1SEM). The total centrality of the nodes changes benignly if network parameters are altered. The error bar for a mean parameter variance of 0 represents the accuracy of the approximation of the total centrality.

Note that sometimes the node names differ slightly from the gene symbols (e.g. RORgt vs. RORγt), because the node names conform to the original publication of the networks or the identifier in the database.

3.3.1 Comparing biological networks

All networks were simulated using the SQDS model with standard parameters (3.2), except for the *A. thaliana* inflorescence network for which Boolean expressions are available (normalized HillCube method).

For some of the networks (e.g. T-helper differentiation, chondrocyte proliferation), alternative continuous models with modified SQDS approaches were proposed in the original publications. Since no biological justification was given for these changes, they

were ignored for this thesis to reduce the number of degrees of freedom in the models and to ensure comparability between the networks.

For the number of runs per centrality each network was simulated see Appendix 7.5.

3.3.1.1 Mouse colon subnetwork

A network describing the differentiation of colon cells from fibroblasts in mice. The networks colon_subnet_1 (10,361 connections), colon_subnet_2 (4,711 connections) and colon_subnet_3 (3,865 connections) from the CellNet website (Cahan, Li 2014) were merged into one large network with 1,310 nodes and 16,742 connections. Duplicate connections were discarded, and loops added to all input nodes.

3.3.1.2 Human T-helper differentiation

A manually modeled network describing the differentiation of human T-helper cells to the effector types Th1, Th2, Th17 and Treg from (Mendoza, Pardo 2010). In the source publication, the network was simulated with an arbitrary SQDS steepness value of 50 for all nodes, artificially improving the reaction of the Foxp3 node (VC sensitivity). For this thesis, the network was simulated with the standard SQDS steepness of 10.

3.3.1.3 Chondrocyte regulation

A manually modeled network describing chondrocyte proliferation and differentiation in human growth plates (Kerkhofs et al. 2012). The network was simulated as a standard SQDS network as opposed to the modified SQDS model from the original publication.

3.3.1.4 *A. thaliana* inflorescence

A manually modeled network describing the differentiation of primordial cells in *Arabidopsis thaliana* during early flower development from the corrigendum (Sánchez-Corrales et al. 2011) to (Sánchez-Corrales et al. 2010).

3.3.1.5 *A. thaliana* immune response

A manually modeled network describing the immune response of *Arabidopsis thaliana* against the gram-negative bacterium Pseudomonas syringae pv tomato DC3000 (Naseem et al. 2012). Self-amplifying loops have been added to the input nodes to better represent external stimuli.

3.3.1.6 *A. thaliana* root stem cell niche

A manually modeled network describing dynamics in the *Arabidopsis thaliana* stem cell niche from Figure 3B in (Azpeitia et al. 2013). Self-amplifying loops have again been added to the input nodes.

3.3.1.7 *S. pombe* (fission yeast) cell cycle

A manually modeled network describing the sequence of cell cycle activation patterns in *Schizosaccharomyces pombe* from Figure 1 B in (Davidich, Bornholdt 2013).

3.3.1.8 *S. cerevisiae* (budding yeast)

A network describing the interaction of transcription factors in *Saccharomyces cerevisiae* compiled from data from the YEASTRACT database (Teixeira et al. 2014). The list of transcription factors with known targets was extracted from the file "RegulationTwoColumnTable_Documented_2013927.tsv.gz" (accessed on 28.03.2014). Interactions were then obtained by querying the database for regulations proved by "Only Expression evidence". To focus on regulatory processes and make the calculations computationally feasible, transcription factors which do not influence other transcription factors and proteins other than transcription factors were excluded from the network, reducing the genome-wide network to a network between those transcription factors.

The latter step is necessary since the whole network including non-regulatory target genes contains more than 200,000 interactions, which exceeds computational limitations on standard hardware.

3.3.1.9 P. aeruginosa

A network describing the interaction of transcription factors in *Pseudomonas aeruginosa* modeled from data by (Galán-Vásquez et al. 2011). The interactions were taken from the sheet "network data" from the additional file "2042-5783-1-3-s1.xls", columns "Regulator (TF or sigma)" and "Target gene". Genes which do not have regulatory influence themselves were again omitted.

3.3.1.10 E. coli

A network describing the interaction of transcription factors in *Escherichia coli* modeled from RegulonDB version 8.5 (Salgado et al. 2013), specifically the first two columns in the file "network_tf_tf.txt" from the "Downloadable Experimental Datasets" section.

3.3.2 *Arabidopsis thaliana* inflorescence

A Boolean model describing flower organ differentiation in *Arabidopsis thaliana* (3.3.1.4) can be used to demonstrate the usefulness of Jimena's ability to quantify basins of attraction and to automatically construct mutant networks.

3.3.2.1 Background

In *A. thaliana*, the formation of the flower begins by the transformation of the inflorescence meristem into the flower meristem which then differentiates into the parts of the flower such as sepals, petals, stamens and carpels.

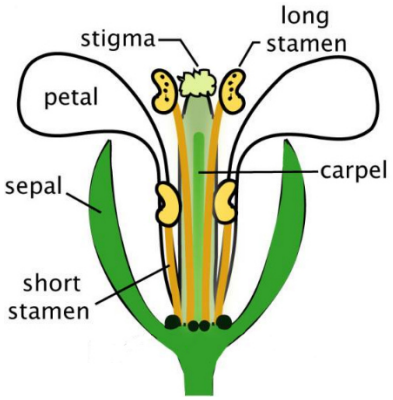


Figure 18: Flower organs in *A. thaliana*
The flower meristems differentiates into sepals, petals, stamens and carpels. (© 2009 Kram et al.)

The original authors analyze the behavior of a discrete network based on their Boolean equations, and of a continuous activator-inhibitor version of the network they simulated using SQUAD (Sánchez-Corrales et al. 2010). Figure 19 shows this simplified version, with

an added amplifying loop at the UFO (UNUSUAL FLOWER ORGANS) node. The Boolean expression of this node is missing from both (Sánchez-Corrales et al. 2010) and its corrigendum (Sánchez-Corrales et al. 2011), and a loop is necessary to reproduce the results in those publications.

In both models, they find that the stable states of the model recover gene expression profiles found in the inflorescence meristem and flower organs.

Four inflorescence states, which they call INF1 – INF4, are characterized by expression of EMF1 (EMBRYONIC FLOWER 1) (Moon 2003) and TFL1 (TERMINAL FLOWER 1) (Shannon, Meeks-Wagner 1993). Once a cell in the inflorescence meristem differentiates to a primordial flower organ cell, these two genes are deactivated and organ specific genes like AP1 (APETALA 1) are expressed (Shannon, Meeks-Wagner 1993). Stable states which the authors call PET1 and PET2, STM1 and STM2, SEP and CAR correspond to the expression profiles of the petal, stamens, sepals and carpels, respectively.

a

	EMF1	TFL1	AP1	AP2	AP3	LFY1	WUS	UFO	SEP	PI	FT	AG	FUL
INF1	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
INF2	1.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
INF3	1.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
INF4	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0
SEP	0.0	0.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0
PET1	0.0	0.0	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0
PET2	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0
CAR	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0
STM1	0.0	0.0	0.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0
STM2	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0

b

	INF1	INF2	INF3	INF4	SEP	PET1	PET2	STM1	STM2	CAR
NHC model (%)	0.005	0.014	0.013	0.035	0.151	0.470	0.017	74.423	8.025	16.848
Discrete model (%)	1.66	1.66	0.88	0.88	9.91	10.05	0.14	37.4	1.15	36.25
SQDS model (%)	4.74	4.77	4.01	4.06	11.01	12.74	1.89	28.46	6.54	21.79

Table 1: Stable states of the *A. thaliana* inflorescence network
a The stable states of the network (HillCube, default parameters) were searched by sampling of the state space starting from 10^6 random initial vectors. The column headings are the names of the network nodes, the row headings are the names of the stable states according to (Sánchez-Corrales et al. 2010). This calculation profits heavily from Jimena's inherent multithreading. **b** Basins of attraction of the stable states. The values for the NHC model have been gained from the calculation in **a**, the values for the discrete model and the SQDS model are from (Sánchez-Corrales et al. 2010).

3.3.2.2 Basins of attraction

Using Jimena, a normalized HillCube version (NHC) of the network based on the original Boolean expressions can now be simulated, avoiding the loss of information that occurs in the transformation to an activator-inhibitor network. Odefy's implementation of HillCubes cannot analyze the network due to several high-degree nodes (Figure 19).

The stable states of the normalized HillCube model (Table 1a) are identical to the ones found for the discrete model in (Sánchez-Corrales et al. 2010).

Interestingly, however, although both models are based on the same Boolean equations, the inflorescence states INF1 – INF4, whose biological validity has been confirmed by gene expression experiments (Espinosa-Soto et al. 2004), are much more unstable in the NHC model, having a combined basin of attraction size of only 0.07% as opposed to 5.1% in the discrete model and 17.6% in the continuous SQDS model (Table 1b). In other words, when simulated from 10^6 random initial states, only 0.06% of the simulations converge to a state corresponding to a non-flowering inflorescence phenotype.

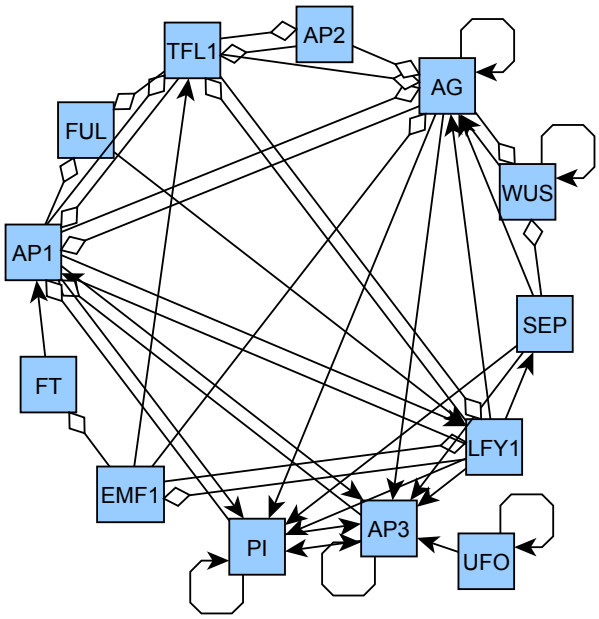


Figure 19: A regulatory network governing *A. thaliana* inflorescence
Activator-inhibitor version of the network proposed by (Sánchez-Corrales et al. 2010) with an added loops to the UFO node (cf. text). Diamond shaped arrow heads represent inhibiting influences.

These results corroborate the hypothesis that the inflorescence attractors are transitory in nature, such that small perturbations lead to progress in plant development and cell differentiation arriving at robust cell states representing flower organs.

3.3.2.3 Mutants

Using active EMF1 and TFL1 nodes (i.e. EMF1 > 0.5 and TFL1 > 0.5) as indicators of inflorescences states, the basins of attraction of the NHC model were then computed assuming single null mutations for the 42 interactions (arrows) of the network (10^4 random start vectors per mutant). The combined basin of attraction size of the inflorescence states of each mutant stayed below 0.5%, except for a removal of the inhibiting influence of AP1 (APETALA1) on TFL1 which leads to a basin size of ~3.5%.

The robustly small size of the basins of attraction of non-flowering cell states is consistent with a reported strong robustness of *A. thaliana* mutants against non-flowering phenotypes (Koornneef et al. 1998).

3.3.3 T-helper cell differentiation

A T-helper cell differentiation network (3.3.1.2) modeling the differentiation of naïve T-helper cells to the subtypes Th1, Th2, Th17 and Treg showcases the utility of the new control centralities and related algorithms.

3.3.3.1 Background

CD4+ T-helper cells play a crucial role in adaptive immunity by orchestrating the immune responses of B-cells, macrophages, neutrophils, eosinophils and other cell types, with almost all these functions being mediated by the secretion of numerous cytokines (Zhu, Paul 2008). Mainly depending on the cytokine environment at the time of the T-cell receptor activation, naïve T-helper cells differentiate to one of several T-helper subtypes, each with its specific cytokine secretion profile. Many of the subtypes are characterized by the activity of a specific master transcriptional regulator gene, which performs its functions via signaling transducer and activator of transcription (STAT) proteins. In turn, the STAT proteins are also involved in the initial activation of the master transcriptional regulator (Luckheeram et al. 2012; Zhu et al. 2010).

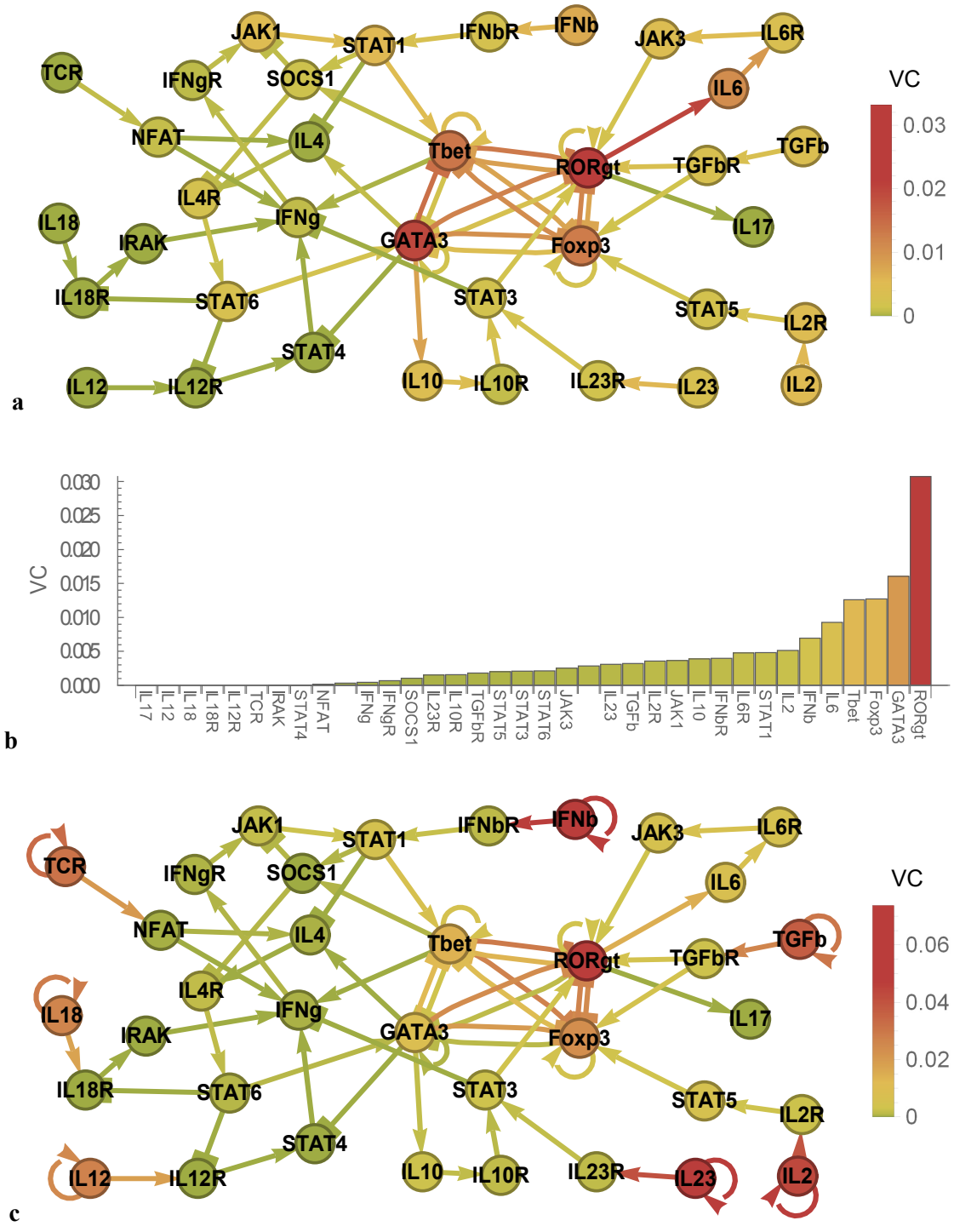


Figure 20: Control in T-helper differentiation
a,b Value centrality in human T-helper cell differentiation (Mendoza, Pardo 2010) (SQDS network) mainly resides in the four master transcriptional regulators Tbet, RORgt, GATA3 and Foxp3. **c** When loops are added to the input nodes in T-helper differentiation, a balance between value centrality of external and internal stimuli is established.

At least four subtypes are clearly discernible (see (Zhu et al. 2010; Luckheeram et al. 2012) for reviews):

Th1 cells, which are involved in the elimination of intracellular pathogens, are characterized by the production of interferon γ (IFN-γ) and lymphotoxin, with many also producing interleukin 2 (IL-2), tumor necrosis factor (TNF, formerly TNFα) and various other cytokines. They can be induced from naïve T-cells by IL-12 in combination with IFN-γ. The master transcriptional regulator of this subtype is T box transcription factor T-bet.

Th2 cells, which contribute to immunity against helminthes and other parasites as well as to allergic inflammation, produce the signature cytokine IL-4, and various others such as, IL-5, IL-9 and IL-13. They can be induced by the presence of both IL-4 and IL-2. The master transcriptional regulator is trans-acting T-cell-specific transcription factor GATA3.

Th17 cells, which are involved in the eradication of extracellular bacteria and fungi, characteristically produce IL-17A and IL-22, among other cytokines like IL-21. The can be induced most efficiently with transforming growth factor β (TGF-β) and IL-21 or IL-6, whereas IL-23 contributes to the maintenance of the subtype via the STAT3 signal transducer. The master transcriptional regulator is RAR-related orphan receptor gamma (RORγt).

Induced T-regulatory cells (iTreg) contribute to self-tolerance and immune response modulation. They can be induced by the simultaneous presence of TGF-β and IL-2. The master transcriptional regulator is forkhead box P3 (Foxp3).

More recent results seem to indicate that follicular T-helper cells, which interact with B-cells within germinal centers contributing for example to B-cell class switching, are a another separate subpopulation (Zhu et al. 2010; Nurieva et al. 2008). The same is true for the Th9 subgroup, which might be crucial for the development of asthma (Luckheeram et al. 2012; Xing et al. 2011; Veldhoen et al. 2008).

3.3.3.2 The role of input loops

In many cases, adding self-amplifying loops to nodes is justified by experimental findings (1.2.4). This is the case for the four central nodes Tbet, GATA3, RORgt and Foxp3 in this T-helper network (Figure 20a), and they were therefore modeled correspondingly by the original authors (Mendoza, Pardo 2010).

From a system's view, it may, however, be reasonable to add artificial self-amplifying loops to the input nodes of the network (like node IL12 in Figure 20c), since these nodes are indispensable links to other parts of the organism-wide regulatory network, representing for example endocrine and paracrine stimuli from other cells. Self-loops enable the nodes to hold state in spite of decay (1.2.5), allowing them to influence the network behavior as external stimuli (cf. Figure 9).

For some networks, such as this T-helper network, different perspectives on node functions are helpful. If self-loops are added to the input nodes, the reaction of the network to external stimuli is examined. Without these loops, the intrinsic behavior without external stimuli is observed, shifting the focus of the analysis to state determining processes within the network.

3.3.3.3 Control centrality

Plotting the value centrality of nodes and connections in the network without input loops (Figure 20a), the values of the nodes Tbet, GATA3, RORgt and Foxp3, whose corresponding genes are the master transcriptional regulators of the T-helper subclasses (3.3.3.1), have the strongest influence on convergence behavior (Figure 20b) and therefore dominate the network. In particular, high value centralities enable them to steer the network in the direction of the subclass they define, while a strong, well-connected central subnetwork between them ensures that only one of the master regulators is active in differentiated T-helper cells (cf. the colors of the connections in Figure 20a).

Interactions between single nodes can also be analyzed. Blocking interleukin 6 (IL6), which is a strong inducer of Th17 cells in conjunction with transforming growth factor-beta (Bettelli et al. 2006), strongly inhibits production of interleukin 17 (IL17) in T-helper cells according to experimental findings (Zhang et al. 2013). Using the ability of the control centralities to measure the influence of one network node on another (2.3.5), the total centrality of all network nodes and connections on IL17 was calculated (Figure 21). The plot shows that the experimental result

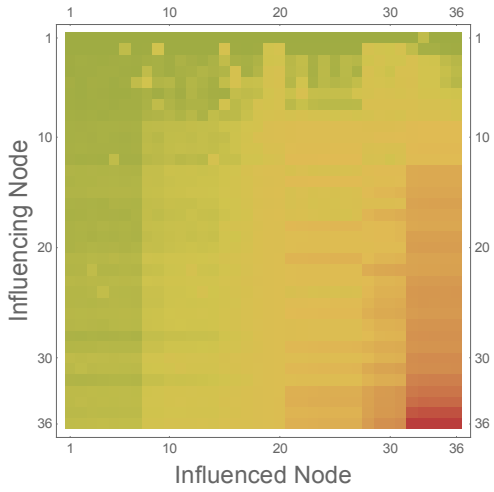


Figure 22: Value centrality influences matrix in T-helper differentiation (without input loops)
Matrix plot indicating the value centrality influence of a node (y-axis) on each other node (x-axis). Nodes on both axes have been ordered by value centrality (y-axis, cf. Figure 20b) and sensitivity (x-axis, Supplementary Material 7.4.1.3).

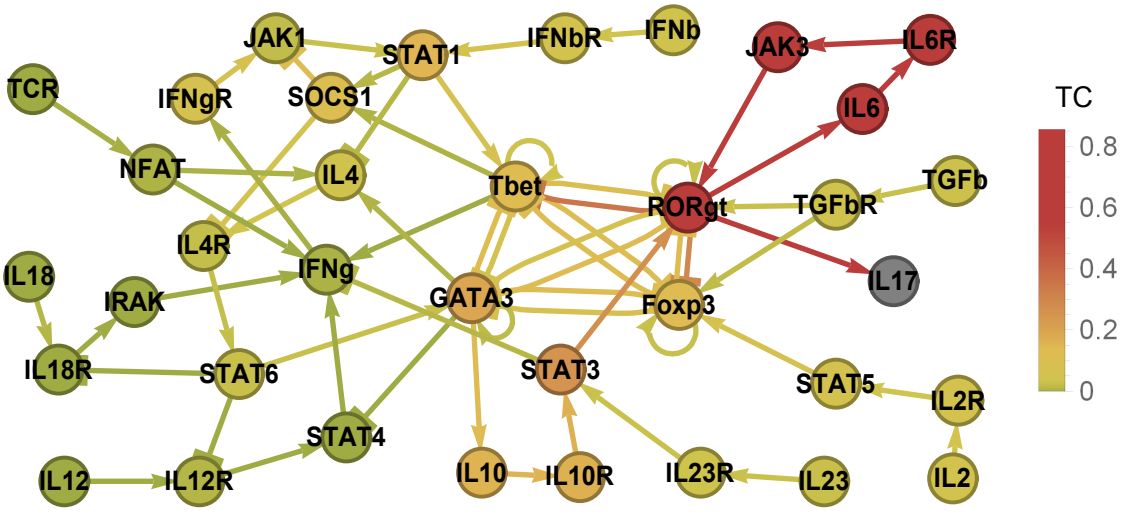


Figure 21: Influences on interleukin 17 in T-helper differentiation
Total centrality on the IL17 node originates in the RORgt→IL6→IL6R→JAK3→RORgt cycle, which is an important target of pharmacological interventions, and in the STAT3 signal transducer.

can be directly related to exceptionally high centralities in the RORgt → IL6 → IL6R → JAK3 → RORgt cycle, with a total centrality influence of IL6 on IL17 of 0.85 (mean of the TC-influences of the nodes outside of this cycle on IL17: 0.05, SD: 0.07).

The node with the strongest influence on IL17 outside of this cycle is STAT3 (TC = 0.25) which is known to be the second main regulator of Th17 cells (Camporeale, Poli 2012).

From a medical perspective, Th17 cells are crucial in the development of multiple autoimmune inflammatory diseases such as rheumatoid arthritis, psoriasis and systemic lupus erythematosus (Zambrano-Zaragoza et al. 2014), and IL-6 (receptor) blockers such as Tocilizumab are already used in clinical routine for example against rheumatoid arthritis (Yao et al. 2014).

Plotting all pairwise influences according to the value centrality (Figure 22) shows that the distribution of the sensitivities is very similar in all rows, indicating that the network is interconnected well enough such that there is only one functional partition which reacts to stimuli from all network nodes.

The situation changes visibly when loops are added to the input nodes (Figure 20c) and parts of the influence of the four central nodes moves to the input nodes, representing crosstalk between external and internal control in T-helper differentiation.

3.3.3.4 Directional value centrality

Directional value centrality can be used to assess whether a network conforms to its modeling assumptions. According to Figure 1 from (Mendoza, Pardo 2010), the authors intended to include the induction of Th1 cells (master regulator: Tbet) by IFN-γ, Th2 cells (GATA3) by IL-4, Th17 cells (RORγt) by TGF-β and IL-6, and Treg cells (Foxp3) by TGF-β. As shown in Figure 23, all these inducers cause a positive response of the

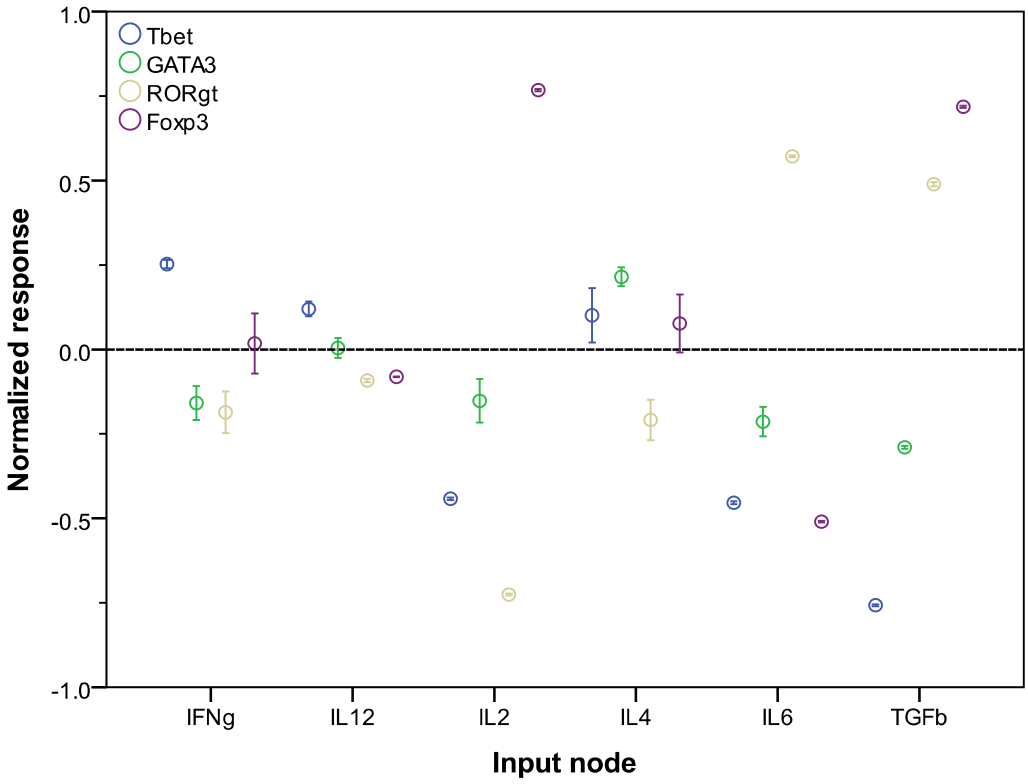


Figure 23: Normalized response of the master transcriptional regulators to 6 input nodes in T-helper differentiation (with input loops)
The directional value centralities (2.4.2) of 6 input nodes on the four master transcriptional regulators (~290,000 runs per data point, error bars 95% CI) were normalized (2.4.3). Points significantly above the dashed line represent activating influences, points below inhibiting ones.

corresponding master regulator, and have a neutral or inhibiting influence on the other three.

The role of IL-2 for the development of Th-2 cells (3.3.3.1) was disregarded in (Mendoza, Pardo 2010), and, correspondingly, the IL2 node does not upregulate GATA3 in this model, leaving room for potential improvements of the model.

Interestingly, many conspicuously strong influences in Figure 23 have already proven to be useful pharmacological targets. As mentioned in 3.3.3.3, the activating influence of IL-6 on RORγt can be blocked in patients with autoimmune diseases to suppress pro-inflammatory Th17 cells, while IL-2 therapy leads to increases in Treg cells counts and corresponding clinical improvements in patients with autoimmune conditions such as hepatitis C associated vasculitis and chronic graft-versus-host disease, which is consistent with the strong activation of Foxp3 by IL-2 in the chart (Koreth et al. 2011; Saadoun et al. 2011).

3.3.3.5 Dynamically connected core network

Applying the MDCN algorithm from 2.4.1 to the T-helper network with the four master regulators (i.e., Tbet, GATA3, RORgt and Foxp3) as significant nodes reveals all dynamically significant connections of these nodes which could be considered the stateful core of the network (Figure 24). All other nodes and connections are merely necessary to feed external stimuli to it, to signal the state of the core network to the output nodes or are irrelevant for the convergence behavior. Note that as in Figure 12, the MDCN in Figure 24 also contains "leaves" such as IL23R $\rightarrow$ STAT3 which are dynamically relevant only by their appearance on the differential equations independently of their value.

Other applications of MDCNs include plausibility checks in network modeling. The algorithm could for example be run on the input and output nodes of a signaling cascade and would essentially check whether all intermediate connections are actually necessary to perform the intended function. By adjusting the Λ threshold, negligible connections could by identified and discarded.

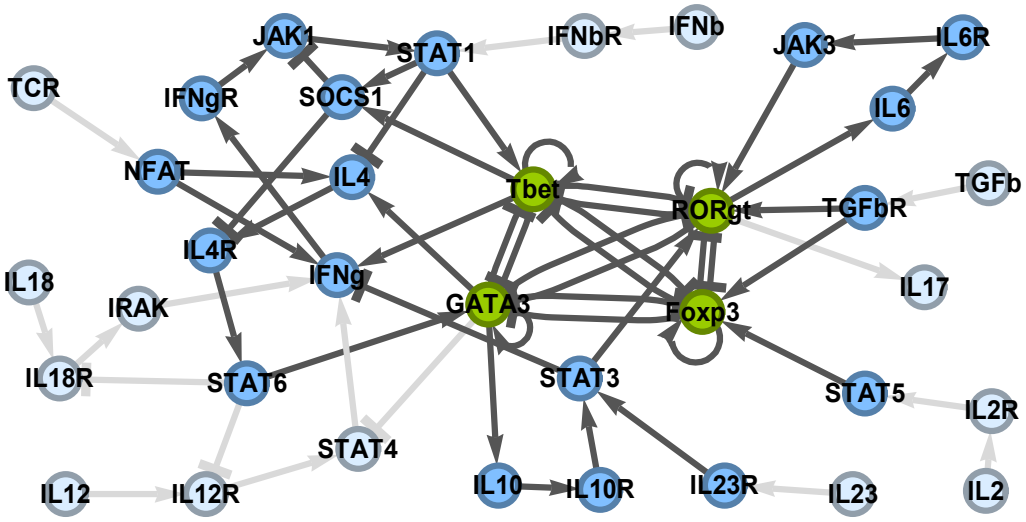


Figure 24: Dynamic core of the T-helper network
MDCN (darker grey, $\Lambda = 10^{-10}$) of the nodes Tbet, GATA3, RORgt and Foxp3 (significant nodes, green color) from the T-helper network (3.3.1.2) (5,000 runs per centrality, 500 for the inner loop in the approximation of the dynamic centrality). Interestingly, there are cycles involving significant nodes that are not relevant for the behavior of these nodes.

3.3.4 Chondrocyte proliferation and differentiation

A network modeling the proliferation and differentiation of chondrocytes in mammalian growth plates (3.3.1.3) demonstrates how the roles of genes (e.g. oncogenes) can be derived from mathematical models using the new control centralities.

3.3.4.1 Background

The elongation of long bones occurs primarily in cartilaginous growth plates in the epiphyses, where proliferating chondrocytes provide a template which is then ossified secondarily by osteoblasts (Figure 25).

Continuously, chondrocytes from a resting zone begin to proliferate, forming long cell columns and building an extracellular cartilage matrix. The chondrocytes then differentiate terminally to non-proliferating hypertrophic chondrocytes, which mineralize

the extracellular matrix before undergoing apoptosis making room for invading blood vessels and ossifying osteoblasts (Solomon et al. 2008).

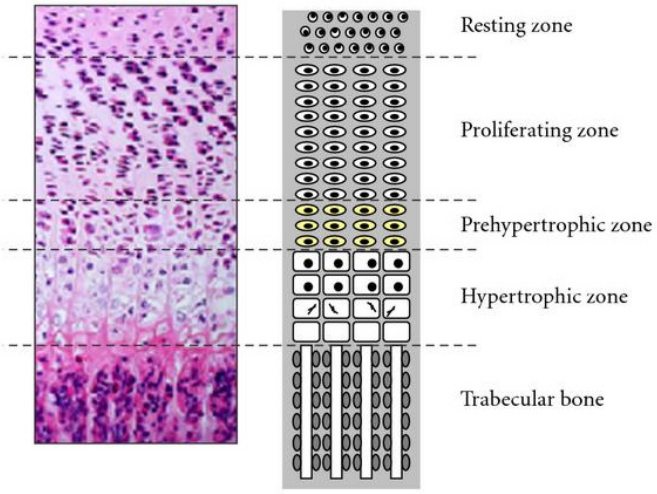


Figure 25: Elongation of long bones in growth plates
In epiphyseal growth plates, chondrocytes from the resting zone (reserve zone) undergo several stages of proliferation, differentiation and hypertrophy elongating the bone, before being replaced by osteoblasts forming the primary spongiosa of the bone. (© 2010 Piróg et al.)

As crucial external signals, parathyroid hormone-related protein (PTHrP) is essential for the maintenance of the proliferating state while the Indian hedgehog signal (Ihh) mediates the transition to the hypertrophic state (Burdan et al. 2009; Minina et al. 2001). The PTHrP/Ihh axis interacts with other signals such as bone morphogenetic proteins (BMPs) (Minina et al. 2001), transforming growth factor beta (TGF-β), Wnt proteins and fibroblast growth factors (FGFs) in regulating sex determining region Y-box 9 (Sox9), the main transcriptional regulator of chondrogenesis (Kronenberg 2003; Burdan et al. 2009).

3.3.4.2 Control centrality

In the corresponding genetic regulatory network (Figure 26), several highly connected nodes and their main effectors exhibit high total centralities (see Appendix 7.4.3.3 for more data). Interestingly, these nodes, such as Sox9 (TC = 0.07, median TC of all nodes $2.7 \cdot 10^{-4}$), BMP (TC = 0.35) and Gli2 (Zinc finger protein GLI2, TC = 0.32), have been experimentally identified as crucial regulators of chondrogenesis and are all implicated in the development of chondrosarcomas (Ho et al. 2009; Boeuf et al. 2012; Tang et al. 2010; Wehrli et al. 2003), corroborating the association between total centrality and the impact of mutations or overexpression. The nodes calculated to have by far the highest

value centrality in the network are the Indian hedgehog signal (extIhh, VC = 0.019, TC = 0.035) and parathyroid hormone-related protein (extPTHrP, VC = 0.010), i.e. the two essential external stimuli of chondrogenesis (Minina et al. 2001) (cf. Appendix 7.4.3.1).

Since therapeutic interventions by drugs or gene therapy exert a permanent influence on the target node, total centrality is suited best to identify potential target genes for intervention. Most of the aforementioned nodes with high total centralities out to be recently validated target nodes. In promising preclinical research, Sox9 gene therapy (Yu et al. 2014) improves the healing of bone-tendon junctions, and Indian hedgehog gene therapy (Steinert et al. 2012) induces chondrogenesis in mesenchymal stem cells. Bone morphogenetic proteins for bone regeneration have already been succesfully tested in

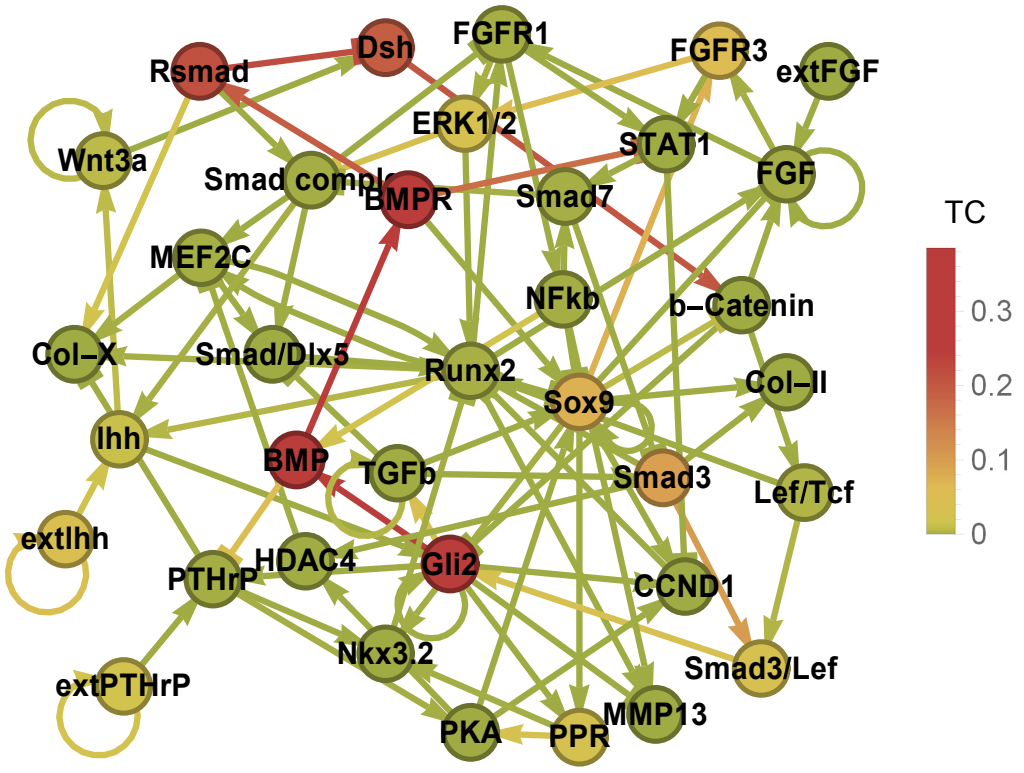


Figure 26: Total centrality in chondrocyte differentiation
Total centrality in chondrocyte proliferation and differentiation (Kerkhofs et al. 2012) reveals a strong central subnetwork around the BMP pathway and the interference of Smad proteins with the Wnt pathway (cf. text).

clinical trials (Carreira et al. 2014) and are increasingly used for musculosceletal disorders, complications of long bone fractures (Lo et al. 2012) or to avoid autogenous bone grafts in reconstructive spine surgery (Even et al. 2012).

Total centrality also elucidates interactions in important signaling pathways such as Wnt and BMP signaling. The Wnt signal is transduced via the Dishevelled (Dsh) protein, which is modulated by the BMP pathways via receptor regulated Smad proteins (Rsmad). The strong, inhibiting total and dynamic centrality of the connection from Rsmad to Dsh (Figure 26 and Appendix 7.4.3.2) therefore heavily interferes with the Wnt signaling pathway, which is in keeping with experimental results showing that BMP-2 strongly inhibits Wnt induced β-Catenin production (Liu et al. 2006).

Node name	Total centrality	Value centrality
cdx2	$1.32 \cdot 10^{-02}$	$6.57 \cdot 10^{-20}$
klf5	$8.46 \cdot 10^{-04}$	$2.76 \cdot 10^{-21}$
hoxa9	$7.93 \cdot 10^{-06}$	$1.85 \cdot 10^{-21}$
hoxb13	$9.85 \cdot 10^{-03}$	$7.39 \cdot 10^{-21}$
vdr	$4.69 \cdot 10^{-03}$	$1.08 \cdot 10^{-20}$
hoxd10	$7.71 \cdot 10^{-04}$	$1.41 \cdot 10^{-21}$
foxd2	$8.21 \cdot 10^{-04}$	$3.00 \cdot 10^{-18}$
hoxa10	$1.04 \cdot 10^{-02}$	$4.27 \cdot 10^{-19}$
satb2	$1.02 \cdot 10^{-03}$	$4.52 \cdot 10^{-21}$
hoxa11	$2.33 \cdot 10^{-05}$	$7.15 \cdot 10^{-21}$
hoxd9	$4.88 \cdot 10^{-03}$	$1.77 \cdot 10^{-20}$
gfi1	$8.30 \cdot 10^{-03}$	$1.77 \cdot 10^{-18}$
klf4	$6.20 \cdot 10^{-06}$	$1.12 \cdot 10^{-21}$
hoxb9	$3.25 \cdot 10^{-05}$	$2.59 \cdot 10^{-21}$
atoh1	$1.54 \cdot 10^{-03}$	$8.67 \cdot 10^{-18}$
ovol2	$1.28 \cdot 10^{-03}$	$1.03 \cdot 10^{-18}$
hoxd13	$2.66 \cdot 10^{-03}$	$7.13 \cdot 10^{-20}$
pitx1	$1.67 \cdot 10^{-03}$	$6.65 \cdot 10^{-20}$
grhl2	$7.83 \cdot 10^{-04}$	$2.93 \cdot 10^{-21}$
ppard	$1.27 \cdot 10^{-04}$	$6.70 \cdot 10^{-21}$

Table 2: Control centrality of the main dysregulators of the mouse colon subnetwork
Total centrality and value centrality of the top twenty transcriptional regulators whose mutation causes dysregulation consistent with aberrant phenotypes in the mouse colon subnetwork. Since the table refers to node names rather than genes, lower case identifiers are used.

3.3.5 Mouse colon subnetwork

The role of centralities is also instructive in a gene network involved in fibroblast differentiation to colon cells in the hindgut (3.3.1.1), studied recently (Morris et al. 2014). In normal liver cells, which develop from the foregut, intestine specific markes such as homeobox transcriptional factor Cdx2 are not expressed in this network. In liver cells artificially induced from mouse embryonic fibroblasts, however, an abnormal expression pattern of the colon network with a partially active hindgut programme is observed.

The authors used their software CellNet (Morris et al. 2014; Müller, Loring 2014) to generate organ specific subnetworks from expression data and reverse engineering. They found 3 subnetworks specific for the colon identity of a cell. Within these networks, they named the top 20 regulators causing this abnormal expression pattern according to a score that quantifies the dysregulation in the expression of target genes caused by mutations in transcriptional regulators (Morris et al. 2014).

For analysis in Jimena, the 3 colon subnetworks were merged into a comprehensive colon network (3.3.1.1). All 20 strong regulators in this network exhibit high total centralities (mean $3.1 \cdot 10^{-3}$, minimum: $6.2 \cdot 10^{-6}$) and the master regulator Cdx2 shows an

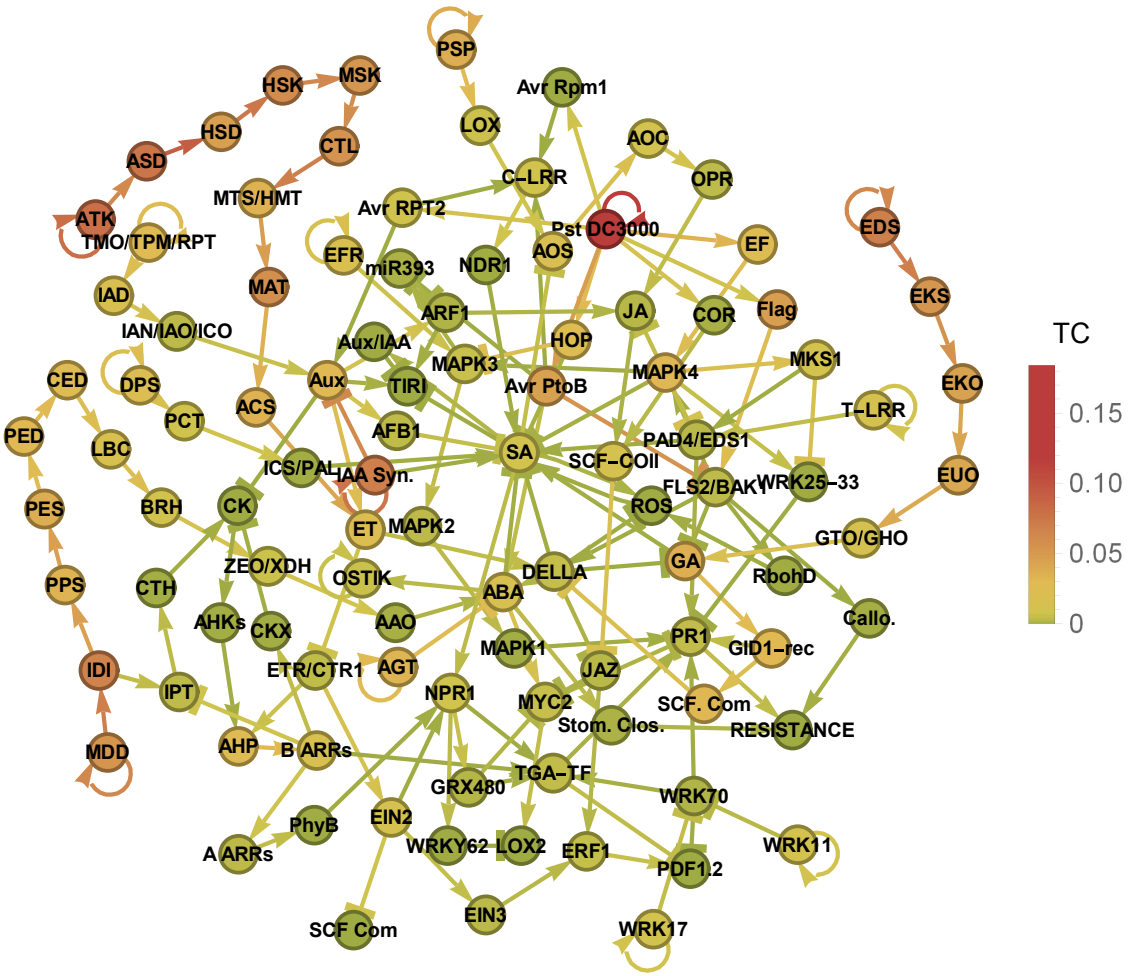


Figure 27: Total centrality in *A. thaliana* immune response
Total centrality of all nodes and connections in a network modeling the immune response of *A. thaliana* against *Pseudomonas syringae*. See Appendix 7.4.2.3 for more data. Very weak (i.e. green) nodes are conspicuously scarce.

exceptionally high total centrality of $1.3 \cdot 10^{-2}$. The value centrality of these nodes is negligible (mean $7.6 \cdot 10^{-19}$), indicating that in this system influence is exerted via dynamic network effects (Table 2).

Jimena thus complements CellNet's functions by revealing that node functions in chondrocyte differentiation are performed by dynamic network control.

3.3.6 *Arabidopsis thaliana* immune response

While plant immunity was long thought to be dominated be a moderate number of small hormonal molecules such as salicylic acid (SA), jasmonic acid (JA) and auxin (AUX) (Robert-Seilaniantz et al. 2011), a new comprehensive regulatory network modeling the immune response of *Arabidopsis thaliana* against the gram negative bacterium Pseudomonas syringae pv. tomato DC3000 (see 3.3.1.5) elucidated the role of other hormones like cytokinin(s) and provided insights into auxin-cytokinin antagonism and cytokinin-SA crosstalk (Naseem et al. 2012).

A plot of the total centrality of the nodes and connections (Figure 27) in this system shows a rather uniform distribution of control indicating that almost all nodes contribute to the behavior of the network (Figure 28a, see Appendix 7.4.5 for more data and graphs). This distribution is to be expected in carefully modeled networks since all nodes have experimentally proven functions in plant immunity which need to be reflected in the behavior of the mathematical model. As expected, the pathogen Pst DC3000 has the greatest influence on network behavior (Figure 28a).

The value centrality matrix (Figure 28b) indicates that in contrast to the T-helper system, the reaction of the network to external stimuli is heavily fragmented, with many partitions reacting to stimuli from only one input node. The matrix also shows that the central effector node RESISTANCE (9^{th} column), which represents the immune response of the plant, is not the most sensitive node, but rather characterized by an evenly distributed sensitivity to changes in a large portion of the network, integrating influences instead of

reacting to just one input node. Mathematically, the RESISTANCE column has the lowest coefficient of variation (3.7) in the network (mean 8.4, SD 2.0).

3.3.6.1 Mutants

To examine the impact of mutants in this network, a discrete Boolean variant of the original was examined. Of the 156 possible single node mutations of the network, only 2 mutations change the number of stable states (from 2 to 1). These are the null mutation of the influence of SA (salicylic acid) on ROS (reactive oxygen species) and of ROS on

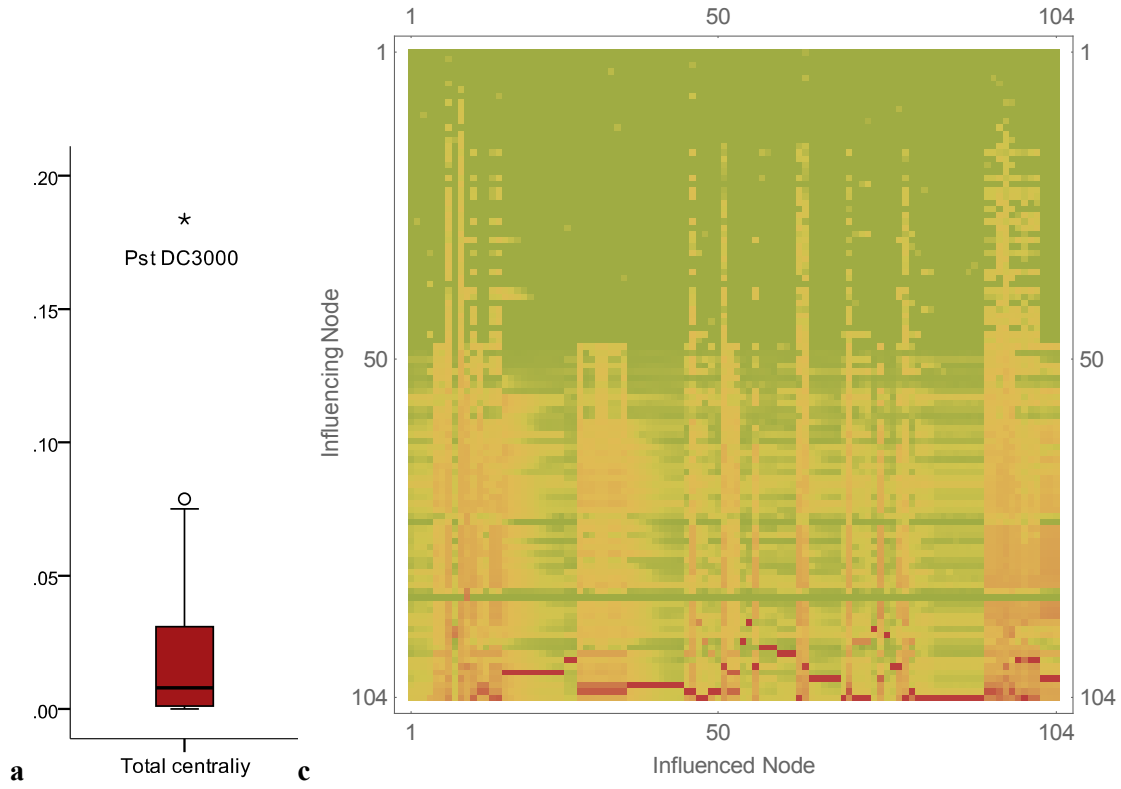


Figure 28: Centralities in *A. thaliana* immune response
a Distribution of total centrality in *A. thaliana* immunity. With a 10%-percentile still at $1.2 \cdot 10^{-6}$ (median $7.9 \cdot 10^{-3}$), almost all nodes contribute to the behavior of the network. The pathogen node Pst DC3000 (representing *Pseudomonas syringae*) is by far the most influential node. **b** Value centrality influences between all nodes in the network (green-yellow-red color scheme). Nodes on both axes have been ordered by value centrality (y-axis) and sensitivity (x-axis). The central output node RESISTANCE is the 9th column in the matrix.

SA, where SA is a key hub node of the network and the small cycle SA → RO →SA is crucial for its ability to save a state.

For all other mutations (n = 154), the changes of the stable states are minor, with only one mutation causing more than four changed nodes in a stable state, namely the removal of ETR/CTR1 (ethylene response / cytosolic serine/threonine kinase constitutive triple response 1) → AHP (Histidine-containing phosphotransmitters) which causes five nodes to change, and most single mutations (n = 142) leading to no change at all.

To check whether the number of stable states increases assuming multiple mutations, the stable states for up to 4 null mutations ($n \approx 2.4 \cdot 10^7$) in the discrete network model were determined, showing that the number of stable states never exceeds 2.

The result of this analysis hints to the robustness of the network against the emergence of new stable states even when up to four connections are removed. Inbuilt robustness is not a rare phenomenon in biological signaling cascades as independently confirmed from experimental data such as promoter recombination trials in *E. coli* (Isalan et al. 2008) or the reported phenotypical robustness of C. albicans against null mutations of transcriptional regulators (Homann et al. 2009). Additional benefits from robustness for this particular signaling cascade include that additional stable states could be detrimental to the latency and efficacy of immune reactions.

Using a single 2.67 GHz CPU core, Jimena constructs and analyzes 2,700 mutants per second in this network, and more than 24,000 mutants per second in the *A. thaliana* inflorescence network (3.3.1.4), which demonstrates its computational efficiency even for complex networks.

3.3.7 Distribution of centralities in biological networks

In biological networks, control centrality is concentrated in an even smaller minority of nodes than in random networks (Table 3, mean Gini(VC) = 0.78, SD = 0.18) indicating that all networks can indeed be controlled by a small minority of the nodes. The networks are sparsely connected and characterized by exceptionally high relative controllabilities compared to random networks (Figure 16).

Manually created networks	**Nodes (Connections)**	**Density**	**RC**	**Gini(VC)**	**Gini(DC)**	**Gini(TC)**
Human T-helper differentiation [with input loops]	36 (62 [69])	1.7 [1.92]	0.113 [0.260]	0.655 [0.713]	0.713 [0.677]	0.642 [0.539]
Mammalian chondrocyte regulation	36 (91)	2.5	0.016	0.951	0.826	0.811
A. thaliana inflorescence	13 (44)	3.4	0.078	0.689	0.749	0.691
A. thaliana immune response (with input loops)	104 (170)	1.6	0.146	0.909	0.684	0.632
A. thaliana root stem cell niche	13 (27)	2.1	0.230	0.820	0.836	0.735
S. pombe cell cycle	12 (31)	2.6	0.206	0.311	0.708	0.576
Transcriptional factor networks						
S. cerevisiae [with input loops]	102 (353 [371])	3.5 [3.6]	0.107 [0.281]	0.949 [0.842]	0.859 [0.829]	0.844 [0.728]
P. aeruginosa [with input loops]	87 (112 [143])	1.3 [1.6]	0.174 [0.347]	0.908 [0.642]	0.881 [0.750]	0.832 [0.526]
E. coli [with input loops]	145 (315 [341])	2.2 [2.4]	0.149 [0.357]	0.935 [0.825]	0.775 [0.691]	0.763 [0.664]

Table 3: Controllability in eukaryotes and prokaryotes
Density, relative controllability (RC) and Gini coefficients of the value centralities (VC), dynamic centralities (DC) and total centralities (TC) in biological networks. See 3.3.1 for the sources of the networks.

4 Discussion

Since a discussion of the results pertaining to specific example was already included in the previous chapter, this chapter aims at a more general synopsis, integrating results from previous sections and outlining potential future developments of the concepts presented.

4.1 Control in biological networks

Defining biological network control is challenging. Various structural and mathematical definitions have been studied (Cornelius et al. 2013; Liu et al. 2011, 2012), but results from in silico network models do not always agree with experimental data, often by overestimating the amount of nodes needed for effective control (Liu et al. 2011; Lee, Young 2013; Müller, Schuppert 2011).

Structural controllability (Liu et al. 2011, 2012) is limited by its focus on network structure instead of network behavior, in spite of recently discussed (Ruths, Ruths 2014b; Campbell et al. 2014) extensions such as statistics on control node patterns ("control profiles") (Ruths, Ruths 2014a), which describe the number of topological patterns called source nodes, sink nodes and dilation points from which structural controllability arises.

This thesis shows how these problems can be overcome by abandoning the strict requirement to drive the network to any desired point in the state space, and focusing instead on network convergence behavior and therefore the fundamental static or dynamic equilibria governing most regulatory processes in biological systems (Cornelius et al. 2013). The new metrics, and their ability to distinguish directly steering nodes (value centrality) from network effects (dynamic centrality), allow for novel insights into biological node functions, network pathways, controllability and robustness, as well as structural motives such as feedback loops or cross-inhibition (3.3).

The intriguingly weak association between value centrality and dynamic centrality found in random Erdős–Rényi and scale-free networks implies that these metrics arise from different topological factors, and self-loops as well as the number of incoming

connections (in-degree) are topological motifs identified to contribute to the difference between these types of control (3.2).

Of course, there are still some shortcomings in the newly defined metrics, and therefore room for improvement. For example, all control centralities consider the behavior of the network for all theoretically possible states of the dynamic system, even if some areas of the state space may never be reached in practice, which could lead to deviations of the results from biological reality. In the future, it may be possible to approximate the actual state space and state trajectories within it from protein and DNA microarray expression data (Huang et al. 2005; Mar, Quackenbush 2009), and restrict the calculation of the integrals in the definitions of the control centralities to this biologically plausible state space, thereby further increasing the biological validity of the metrics.

Several analyses show that large parts of the value centrality are situated at the network borders (e.g. 3.3.3), where external stimuli are fed into the system (e.g. via G-coupled transmembrane receptors as well as other important drug targets). With increasing integration of regulatory systems into larger, more comprehensive networks, there should be a tendency that value centrality disappears from the former network borders and is superseded by more dynamic centrality and control.

Another contentious point regarding centralities in biological networks is the connection to genomic indices measuring positive evolutionary selection. While some studies suggested a negative correlation between the degree of the node (a simple type of structural centrality) and evolutionary speed (Fraser et al. 2002; Coulombe-Huntington, Xia 2012), others claim the opposite (Jovelin, Phillips 2009). As soon as high quality data on evolutionary rates such as Ka/Ks ratios (Hurst 2002) becomes available for model organisms, it may be fruitful to compare them to the control centralities in this thesis, although it may prove difficult to map each network node to specific gene since nodes often represent families of similar proteins or proteins comprised of several genetically independent subunits.

4.2 Controllability and robustness

In the analyses of random as well as real-world networks several common themes emerged. Perhaps the most salient is the equilibrium of mutational robustness and functional flexibility which shapes the behavior of biological networks (3.2, 3.3.7). On the one hand, there is evolutionary selection towards systems which are less susceptible to nocuous mutations (Pu et al. 2012; Wagner 2005; Macneil, Walhout 2011) which may lead to cancer and hereditary diseases. On the other hand, the system must be flexible enough to react swiftly to external stimuli and to exhibit evolvability enabling long term adaptations (Poole et al. 2003).

In the control centrality metrics established in this thesis, this trade-off manifests itself in high ratios between value centrality and total centrality (relative controllability) found in biological networks (3.2, 3.3.7). Analysis of large numbers of random networks helped to identify topological factors increasing this ratio. One of these aspects, which might arise from this evolutionary pressure, are self-regulating loops which noticeably increase relative controllability, explaining for example their prevalence in the biological networks from chapter 3.3.

Another ubiquitous factor contributing to high relative controllabilities are moderate network densities around 2.5, i.e. each node experiences direct influence from an average of 2.5 other nodes, for example as transcription factors. It should be noted, however , that this results refers only to strong interactions between nodes, and it is probable that in biological networks large numbers of weaker, modulating influences are present which are not captured by current network models. Recently, more and more of these weaker influences have been identified experimentally and included in interactions databases (1.2.6), often disregarding information on the weakness of the influences. Naïvely including all these connections in network models as influences of equal strength would sharply increase network densities, thus stiffening network behavior and artificially decreasing relative controllability. In the future, it will therefore be essential to maintain

an interrupted flow of information from quantitative experimental results to the databases and from there to mathematical models.

4.3 Reverse engineering biological networks

To avoid the impression that regulatory networks need to be tailored specifically to the newly developed control centralities, in this thesis only networks from external researchers and networks created automatically from external databases were used. But the reverse is also conceivable.

When modeling a regulatory network, it is usually known, for example from knockout experiments, which nodes and connections influence the network the most. It may thus be advantageous to gauge the centralities while the model is still developing, identifying network structures which are unrealistically strong or weak. For a given function of a regulatory network, calculating the minimal dynamically connected network (2.4.1) might be a straightforward tool to check whether all connections currently in the model are actually functionally active. Plausibility checks are also possible by determining the directional value centrality (2.4.2) between known master regulators and important external effectors (3.3.3.4).

One step further, it may even be possible to directly control the centralities in a model using well-defined network manipulations. For example, one could explore a range of possible model modifications which strengthen the centrality of a master regulator which is too weak in the current draft of model, and check whether there is biological evidence for one of them. (Nicosia et al. 2012) pioneered techniques to modify certain non-control centralities by altering the relative weight of influences in the network, and (Pan, Li 2014) demonstrated how to regulate the structural control centralities from (Liu et al. 2012). While these centralities based on network topology can be controlled rather easily by altering said topology, it is unlikely that a deterministic algorithm will be found to exactly tune the control centralities in this thesis. Stochastic or evolutionary algorithms to this end are, however, conceivable and an interesting area for further study.

Another interesting evolutionary modeling algorithm could arise from the network behavior comparison $\Delta(N_1, N_2)$ from 2.3.1. Note that one of the networks in the comparison does not have to be a mathematical model, but it could also be an experimental biological setting were the convergence of the real-world biological network can be examined from different initial states. It is then possible to evolutionarily modify the topology and numerical parameters of a primordial mathematical model such that the difference Δ between reality and model is minimized. This approach is similar to the fitting of model parameters to time-series data shown by (Di Cara et al. 2007), but includes changes to the topology and is also possible in the absence of time-series data, for example in systems that converge very quickly. A simplified version of this network comparison method, which relies solely on the stable states instead of taking the basins of attraction into account, has also been proposed by (Qian, Dougherty 2013).

4.4 Medical implications

Although many of the new methods proposed in this thesis have been developed to elucidate basic mechanisms of control in biological systems, several medical applications have been mentioned.

With regard to (potential) pharmacological targets in human regulatory networks, two complementary approaches were used. The one-time stimulus necessary to induce the differentiation of naïve T-helper cells for example to non-inflammatory Treg cells is best represented by value centrality (3.3.3.3), whose directional counterpart can even predict whether a pharmacological activation or blockage is required (3.3.3.4). Permanent influences such as Sox9 gene therapy in chondrocytes are investigated best using total centrality (3.3.4.2).

It is crucial to keep in mind that these are exploratory results in manually modelled networks, and further research in future high-quality genome-wide networks is needed to validate the concept.

Another concept with immediate medical applications is the regulatory robustness of genetic networks, discussed here for *A. thaliana* immune responses (3.3.6.1) and theoretically in the form of relative controllability (3.2). Recent results suggest that a simplified Boolean human signaling network from (Helikar et al. 2008) is separated into highly controllable core networks (called "evolvable" by the authors), and other more robust parts. Intriguingly, new drugs approved by the US Food and Drug Administration are found mainly in the controllable parts (Kim et al. 2014).

As additional results, (Kim et al. 2014) also observe that the controllable partitions in this biological network are conspicuously rich in feedback loops and exhibit a more scale-free node degree distribution, which confirms theoretical results from this thesis (3.2).

From a medical perspective, the robustness discussed here is not always beneficial. For example, treatment resistance against drugs targeting the epidermal growth factor receptor (EGFR) pathway in glioblastoma patients has been linked to signaling robustness of glioblastoma cells (Azuaje et al. 2015), and the development of resistance against EGFR-targeted drugs in lung cancer has been attributed to the activation of alternative receptor tyrosine kinase pathways (Niederst, Engelman 2013). Understanding the mechanism behind this robustness by analyzing dynamic control could prove fundamental to breaking drug resistance with a combined therapy regimen.

Apart from the examples in this thesis, Jimena is used in ongoing research at the Chair of Bioinformatics Würzburg. Current projects include the modeling of oncogenic mutations in cancer cells and their modulation by targeted pharmacological interventions such as EGFR inhibition by Gefitinib, and the effects of platelets stimulating the response of dendritic cells against Aspergillus fumigatus infections. Jimena's utility in analyzing differentiating and proliferating cells (as shown here for T-helper cells and chondrocytes, 3.3.3 and 3.3.4) is used to elucidate the role of stem cells in osteogenesis and chondrogenesis, and the role of extracellular-signal-regulated kinases (ERKs) in heart hypertrophy.

5 Summary

Biological systems such as cells or whole organisms are governed by complex regulatory networks of transcription factors, hormones and other regulators which determine the behavior of the system depending on internal and external stimuli. In mathematical models of these networks, genes are represented by interacting “nodes” whose “value” represents the activity of the gene.

Control processes in these regulatory networks are challenging to elucidate and quantify. Previous control centrality metrics, which aim to mathematically capture the ability of individual nodes to control biological systems, have been found to suffer from problems regarding biological plausibility.

This thesis presents a new approach to control centrality in biological networks. Three types of network control are distinguished: Total control centrality quantifies the impact of gene mutations and identifies potential pharmacological targets such as genes involved in oncogenesis (e.g. zinc finger protein GLI2 or bone morphogenetic proteins in chondrocytes). Dynamic control centrality describes relaying functions as observed in signaling cascades (e.g control in mouse colon stem cells). Value control centrality measures the direct influence of the value of the node on the network (e.g. Indian hedgehog as an essential regulator of proliferation in chondrocytes). Well-defined network manipulations define all three centralities not only for nodes, but also for the interactions between them, enabling detailed insights into network pathways.

The calculation of the new metrics is made possible by substantial computational improvements in the simulation algorithms for several widely used mathematical modeling paradigms for genetic regulatory networks, which are implemented in the regulatory network simulation framework Jimena created for this thesis.

Applying the new metrics to biological networks and artificial random networks shows how these mathematical concepts correspond to experimentally verified gene functions and signaling pathways in immunity and cell differentiation. In contrast to controversial

previous results even from the Barabási group, all results indicate that the ability to control biological networks resides in only few driver nodes characterized by a high number of connections to the rest of the network. Autoregulatory loops strongly increase the controllability of the network, i.e. its ability to control itself, and biological networks are characterized by high controllability in conjunction with high robustness against mutations, a combination that can be achieved best in sparsely connected networks with densities (i.e. connections to nodes ratios) around 2.0 - 3.0.

The new concepts are thus considerably narrowing the gap between network science and biology and can be used in various areas such as system modeling, plausibility trials and system analyses.

Medical applications discussed in this thesis include the search for oncogenes and pharmacological targets, as well their functional characterization.

5.1 Zusammenfassung

Biologische Systeme wie Zellen aber auch ganze Organismen werden durch ein komplexes Netzwerk von Transkriptionsfaktoren, Hormonen und anderen Regulatoren kontrolliert, welche das Verhalten des Systems in Abhängigkeit von internen und externen Einflüssen steuern. In mathematischen Modellen dieser Netzwerke werden Gene durch „Knoten“ repräsentiert, deren „Wert“ die Aktivität des Gens wiederspiegelt.

Kontrollvorgänge in diesen Regulationsnetzwerken sind schwierig zu quantifizieren. Existierende Maße für die Kontrollzentralität, d.h. die Fähigkeit einzelner Knoten biologische Systeme zu kontrollieren, zeigen vor allem Probleme mit der biologischen Plausibilität der Ergebnisse.

Diese Dissertation stellt eine neue Definition der Kontrollzentralität vor. Dabei werden drei Typen der Kontrollzentralität unterschieden: Totale Kontrollzentralität quantifiziert den Einfluss von Mutationen eines Gens und hilft mögliche pharmakologische Ziele wie etwa Onkogene (z. B. das Zinkfingerprotein GLI2 oder Bone Morphogenetic Proteins in Chondrozyten) zu identifizieren. Dynamische Kontrollzentralität beschreibt signalweiterleitende Funktionen in Signalkaskaden (z. B. in Kontrollprozessen in Stammzellen des Mauskolons). Wert-Kontrollzentralität misst den Einfluss des Werts des Knotens (zum Beispiel die Rolle von Indian hedgehog als essentieller Regulator der Chondrozytenproliferation). Durch gezielte Manipulation von Netzwerken können die Zentralitäten nicht nur für Knoten, sondern auch für die Interaktionen zwischen ihnen bestimmt werden, was detaillierte Einblicke in Netzwerkpfade erlaubt.

Möglich wird die Berechnung der neuen Maße durch substantielle Verbesserungen der Simulationsalgorithmen mehrerer häufig verwendeter mathematischer Muster für Genregulationsnetzwerke, welche in der für diese Dissertation entwickelten Software Jimena implementiert wurden.

Durch die Anwendung der neuen Metriken auf biologische Netzwerke und künstliche Zufallsnetzwerke kann gezeigt werden, dass die mathematischen Konzepte experimentell bestätigte Funktionen von Genen und Signalpfaden im Immunsystem und der

Zelldifferenzierung korrekt wiedergeben. Im Gegensatz zu umstrittenen Ergebnissen der Forschungsgruppe Barabási zeigt sich hier, dass die Fähigkeit, biologische Netzwerke zu kontrollieren, in nur wenigen Knoten konzentriert ist, welche sich vor allem durch viele Verbindungen zum Rest des Netzwerks auszeichnen. Knoten, welche ihre eigene Expression beeinflussen, steigern die Fähigkeit eines Netzwerkes sich selbst zu kontrollieren (Kontrollierbarkeit), und biologische Netzwerke zeichnen sich durch hohe Kontrollierbarkeit bei gleichzeitig hoher Resistenz gegenüber Mutationen aus. Diese Kombination kann am besten durch eher schwach verbundene Netzwerke erreicht werden, bei denen auf einen Knoten nur etwa 2 bis 3 Verbindungen kommen.

Die neuen Konzepte schlagen so eine Brücke zwischen Netzwerkwissenschaften und Biologie, und sind in einer Vielzahl von Gebieten wie der Modellierung von Systemen sowie der Überprüfung ihrer Plausibilität und ihrer Analyse anwendbar.

Medizinische Anwendungen, auf welche in dieser Dissertation eingegangen wird, sind zum Beispiel die Suche nach Onkogenen und pharmakologischen Zielen, aber auch deren funktionelle Analyse.

Publication bibliography

1. Adachi, Megumi; Monteggia, Lisa M. (2014): Decoding transcriptional repressor complexes in the adult central nervous system. In *Neuropharmacology* 80, pp. 45–52. DOI: 10.1016/j.neuropharm.2013.12.024.

2. Albert, István; Thakar, Juilee; Li, Song; Zhang, Ranran; Albert, Réka (2008): Boolean network simulations for life scientists. In *Source Code Biol Med* 3, p. 16. DOI: 10.1186/1751-0473-3-16.

3. Albert, Réka (2002): Statistical mechanics of complex networks. In *Rev. Mod. Phys.* 74 (1), pp. 47–97. DOI: 10.1103/RevModPhys.74.47.

4. Azpeitia, Eugenio; Weinstein, Nathan; Benítez, Mariana; Mendoza, Luis; Alvarez-Buylla, Elena R. (2013): Finding Missing Interactions of the Arabidopsis thaliana Root Stem Cell Niche Gene Regulatory Network. In *Front Plant Sci* 4, p. 110. DOI: 10.3389/fpls.2013.00110.

5. Azuaje, Francisco; Tiemann, Katja; Niclou, Simone P. (2015): Therapeutic control and resistance of the EGFR-driven signaling network in glioblastoma. In *Cell communication and signaling : CCS* 13, p. 23.

6. Barabási, Albert-László (2003): Linked. How everything is connected to everything else and what it means for business, science, and everyday life. New York: Plume.

7. Barabási, Albert-László; Albert, Réka (1999): Emergence of Scaling in Random Networks. In *Science* 286 (5439), pp. 509–512. DOI: 10.1126/science.286.5439.509.

8. Ben-Ari, Mordechai (2012): Mathematical Logic for Computer Science. London: Springer London.

9. Bender, Carl M.; Orszag, Steven A. (1999): Advanced mathematical methods for scientists and engineers. New York: Springer.

10. Bertrand, Daniel; Gopalakrishnan, Murali (2007): Allosteric modulation of nicotinic acetylcholine receptors. In *Biochem. Pharmacol.* 74 (8), pp. 1155–1163. DOI: 10.1016/j.bcp.2007.07.011.

11. Bettelli, Estelle; Carrier, Yijun; Gao, Wenda; Korn, Thomas; Strom, Terry B.; Oukka, Mohamed et al. (2006): Reciprocal developmental pathways for the generation of pathogenic effector TH17 and regulatory T cells. In *Nature* 441 (7090), pp. 235–238. DOI: 10.1038/nature04753.

12. Boeuf, Stephane; Bovée, Judith VMG; Lehner, Burkhard; van den Akker, Brendy; van Ruler, Maayke; Cleton-Jansen, Anne-Marie; Richter, Wiltrud (2012): BMP and TGFbeta pathways in human central chondrosarcoma: enhanced endoglin and Smad 1 signaling in high grade tumors. In *BMC Cancer* 12 (1), p. 488. DOI: 10.1186/1471-2407-12-488.

13. Bonacich, Phillip; Lloyd, Paulette (2001): Eigenvector-like measures of centrality for asymmetric relations. In *Social Networks* 23 (3), pp. 191–201. DOI: 10.1016/S0378-8733(01)00038-7.

14. Bovolenta, Luiz A.; Acencio, Marcio L.; Lemke, Ney (2012): HTRIdb: an open-access database for experimentally verified human transcriptional regulation interactions. In *BMC Genomics* 13, p. 405. DOI: 10.1186/1471-2164-13-405.

15. Bryant (1986): Graph-Based Algorithms for Boolean Function Manipulation. In *IEEE Trans. Comput.* C-35 (8), pp. 677–691. DOI: 10.1109/TC.1986.1676819.

16. Burdan, Franciszek; Szumiło, Justyna; Korobowicz, Agnieszka; Farooquee, Rabia; Patel, Sagar; Patel, Ankit et al. (2009): Morphology and physiology of the epiphyseal growth plate. In *Folia histochemica et cytobiologica / Polish Academy of Sciences, Polish Histochemical and Cytochemical Society* 47 (1), pp. 5–16. DOI: 10.2478/v10042-009-0007-1.

17. Burg, Klemens; Haf, Herbert; Meister, Andreas (2009): Gewöhnliche Differentialgleichungen, Distributionen, Integraltransformationen. Mit 12 Tabellen

und 85 Aufgaben. 5., überarb. und erw. Aufl. Wiesbaden: Vieweg + Teubner (Studium, ; Bd. 3).

18. Cahan, Patrick; Li, Hu (2014): Mouse GRNs. Available online at http://cellnet.hms.harvard.edu/grn/mouse, checked on 10/14/2014.

19. Campbell, Colin; Shea, Katriona; Albert, Réka (2014): Network models. Comment on "Control profiles of complex networks". In *Science (New York, N.Y.)* 346 (6209), p. 561. DOI: 10.1126/science.1256492.

20. Camporeale, Annalisa; Poli, Valeria (2012): IL-6, IL-17 and STAT3: a holy trinity in auto-immunity? In *Frontiers in bioscience (Landmark edition)* 17, pp. 2306–2326.

21. Carreira, Ana Claudia; Alves, Gutemberg Gomes; Zambuzzi, William Fernando; Sogayar, Mari Cleide; Granjeiro, José Mauro (2014): Bone Morphogenetic Proteins: Structure, biological function and therapeutic applications. In *Archives of Biochemistry and Biophysics. DOI:* 10.1016/j.abb.2014.07.011.

22. Chatr-Aryamontri, Andrew; Breitkreutz, Bobby-Joe; Heinicke, Sven; Boucher, Lorrie; Winter, Andrew; Stark, Chris et al. (2013): The BioGRID interaction database: 2013 update. In *Nucleic Acids Res.* 41 (Database issue), pp. D816-23. DOI: 10.1093/nar/gks1158.

23. Cheng, Heung-Chin; Qi, Robert Z.; Paudel, Hemant; Zhu, Hong-Jian (2011): Regulation and function of protein kinases and phosphatases. In *Enzyme Res* 2011, p. 794089. DOI: 10.4061/2011/794089.

24. Clauset, Aaron; Shalizi, Cosma Rohilla; Newman, M. E. J. (2009): Power-Law Distributions in Empirical Data. In *SIAM Rev.* 51 (4), pp. 661–703. DOI: 10.1137/070710111.

25. Cornelius, Sean P.; Kath, William L.; Motter, Adilson E. (2013): Realistic control of network dynamics. In *Nat Commun* 4, p. 1942. DOI: 10.1038/ncomms2939.

26. Coulombe-Huntington, Jasmin; Xia, Yu (2012): Regulatory network structure as a dominant determinant of transcription factor evolutionary rate. In *PLoS computational biology* 8 (10), pp. e1002734. DOI: 10.1371/journal.pcbi.1002734.

27. Davidich, Maria I.; Bornholdt, Stefan (2013): Boolean network model predicts knockout mutant phenotypes of fission yeast. In *PLoS ONE* 8 (9), pp. e71786. DOI: 10.1371/journal.pone.0071786.

28. Di Cara, Alessandro; Garg, Abhishek; Micheli, Giovanni de; Xenarios, Ioannis; Mendoza, Luis (2007): Dynamic simulation of regulatory networks using SQUAD. In *BMC Bioinformatics* 8, p. 462. DOI: 10.1186/1471-2105-8-462.

29. Di Croce, Luciano; Helin, Kristian (2013): Transcriptional regulation by Polycomb group proteins. In *Nature structural & molecular biology* 20 (10), pp. 1147–1155. DOI: 10.1038/nsmb.2669.

30. Dixon, Philip; Weiner, Jacob; Mitchell-Olds, Thomas; Woodley, Robert (1988): Erratum to 'Bootstrapping the Gini Coefficient of Inequality'. In *Ecology* 69 (4), p. 1307. DOI: 10.2307/1941290.

31. Dixon, Philip M.; Weiner, Jacob; Mitchell-Olds, Thomas; Woodley, Robert (1987): Bootstrapping the Gini Coefficient of Inequality. In *Ecology* 68 (5), p. 1548. DOI: 10.2307/1939238.

32. Dolev, Shlomi; Elovici, Yuval; Puzis, Rami (2010): Routing betweenness centrality. In *J. ACM* 57 (4), pp. 1–27. DOI: 10.1145/1734213.1734219.

33. Eid, Michael; Gollwitzer, Mario; Schmitt, Manfred (2011): Statistik und Forschungsmethoden. Lehrbuch; mit Online-Materialien. 2., korr. Aufl. Weinheim, Basel: Beltz.

34. Erdős, Paul; Rényi, Alfréd (1959): On Random Graphs I. In *Publicationes Mathematicae* 6, pp. 290–297.

35. Espinosa-Soto, Carlos; Padilla-Longoria, Pablo; Alvarez-Buylla, Elena R. (2004): A gene regulatory network model for cell-fate determination during Arabidopsis thaliana flower development that is robust and recovers experimental gene expression profiles. In *Plant Cell* 16 (11), pp. 2923–2939. DOI: 10.1105/tpc.104.021725.

36. Even, Jesse; Eskander, Mark; Kang, James (2012): Bone morphogenetic protein in spine surgery: current and future uses. In *The Journal of the American Academy of Orthopaedic Surgeons* 20 (9), pp. 547–552.

37. Franceschini, Andrea; Szklarczyk, Damian; Frankild, Sune; Kuhn, Michael; Simonovic, Milan; Roth, Alexander et al. (2013): STRING v9.1: protein-protein interaction networks, with increased coverage and integration. In *Nucleic Acids Res.* 41 (Database issue), pp. D808-15. DOI: 10.1093/nar/gks1094.

38. Fraser, Hunter B.; Hirsh, Aaron E.; Steinmetz, Lars M.; Scharfe, Curt; Feldman, Marcus W. (2002): Evolutionary rate in the protein interaction network. In *Science (New York, N.Y.)* 296 (5568), pp. 750–752. DOI: 10.1126/science.1068696.

39. Freeman, Linton C. (1978): Centrality in social networks conceptual clarification. In *Social Networks* 1 (3), pp. 215–239. DOI: 10.1016/0378-8733(78)90021-7.

40. Fulton, Debra L.; Sundararajan, Saravanan; Badis, Gwenael; Hughes, Timothy R.; Wasserman, Wyeth W.; Roach, Jared C.; Sladek, Rob (2009): TFCat: the curated catalog of mouse and human transcription factors. In *Genome Biol.* 10 (3), pp. R29. DOI: 10.1186/gb-2009-10-3-r29.

41. Galán-Vásquez, Edgardo; Luna, Beatriz; Martínez-Antonio, Agustino (2011): The Regulatory Network of Pseudomonas aeruginosa. In *Microb Inform Exp* 1 (1), p. 3. DOI: 10.1186/2042-5783-1-3.

42. Garg, Abhishek; Di Cara, Alessandro; Xenarios, Ioannis; Mendoza, Luis; Micheli, Giovanni de (2008): Synchronous versus asynchronous modeling of gene regulatory networks. In *Bioinformatics* 24 (17), pp. 1917–1925. DOI: 10.1093/bioinformatics/btn336.

43. Garg, Abhishek; Xenarios, Ioannis; Mendoza, Luis; DeMicheli, Giovanni (2007): An efficient method for dynamic analysis of gene regulatory networks and in silico gene perturbation experiments. In : Proceedings of the 11th annual international conference on Research in computational molecular biology. Oakland, CA, USA: Springer-Verlag, pp. 62–76.

44. Garneau, Nicole L.; Wilusz, Jeffrey; Wilusz, Carol J. (2007): The highways and byways of mRNA decay. In *Nat. Rev. Mol. Cell Biol.* 8 (2), pp. 113–126. DOI: 10.1038/nrm2104.

45. Gershenson, Carlos (2004): Introduction to Random Boolean Network. In *Workshop and Tutorial Proceedings, Ninth International Conference on the Simulation and Synthesis of Living Systems (ALife IX)*, pp. 160–173.

46. Helikar, Tomás; Konvalina, John; Heidel, Jack; Rogers, Jim A. (2008): Emergent decision-making in biological signal transduction networks. In *Proceedings of the National Academy of Sciences of the United States of America* 105 (6), pp. 1913–1918.

47. Ho, Louisa; Stojanovski, Aneta; Whetstone, Heather; Wei, Qing Xia; Mau, Elaine; Wunder, Jay S.; Alman, Benjamin (2009): Gli2 and p53 Cooperate to Regulate IGFBP-3- Mediated Chondrocyte Apoptosis in the Progression from Benign to Malignant Cartilage Tumors. In *Cancer Cell* 16 (2), pp. 126–136. DOI: 10.1016/j.ccr.2009.05.013.

48. Homann, Oliver R.; Dea, Jeanselle; Noble, Suzanne M.; Johnson, Alexander D. (2009): A phenotypic profile of the Candida albicans regulatory network. In *PLoS Genet.* 5 (12), pp. e1000783. DOI: 10.1371/journal.pgen.1000783.

49. Huang, Sui; Eichler, Gabriel; Bar-Yam, Yaneer; Ingber, Donald E. (2005): Cell fates as high-dimensional attractor states of a complex gene regulatory network. In *Physical review letters* 94 (12), p. 128701.

50. Hurst, Laurence D. (2002): The Ka/Ks ratio: diagnosing the form of sequence evolution. In *Trends in Genetics* 18 (9), pp. 486–487. DOI: 10.1016/S0168-9525(02)02722-1.

51. IEEE (2008): IEEE standard for floating-point arithmetic. New York, NY: Institute of Electrical and Electronics Engineers (IEEE Std 754-2008).

52. Isalan, Mark; Lemerle, Caroline; Michalodimitrakis, Konstantinos; Horn, Carsten; Beltrao, Pedro; Raineri, Emanuele et al. (2008): Evolvability and hierarchy in

rewired bacterial gene networks. In *Nature* 452 (7189), pp. 840–845. DOI: 10.1038/nature06847.

53. Jovelin, Richard; Phillips, Patrick C. (2009): Evolutionary rates and centrality in the yeast gene regulatory network. In *Genome biology* 10 (4), pp. R35. DOI: 10.1186/gb-2009-10-4-r35.

54. Kalman, R. E. (1963): Mathematical Description of Linear Dynamical Systems. In *Journal of the Society for Industrial and Applied Mathematics Series A Control* 1 (2), pp. 152–192. DOI: 10.1137/0301010.

55. Kanehisa, Minoru; Goto, Susumu; Sato, Yoko; Furumichi, Miho; Tanabe, Mao (2012): KEGG for integration and interpretation of large-scale molecular data sets. In *Nucleic Acids Res.* 40 (Database issue), pp. D109-14. DOI: 10.1093/nar/gkr988.

56. Karl, Stefan (2013): Jimena. A Java genetic regulatory network simulation framework. Available online at http://stefan-karl.de/jimena/, checked on 10/30/2014.

57. Karl, Stefan; Dandekar, Thomas (2013): Jimena: efficient computing and system state identification for genetic regulatory networks. In *BMC Bioinformatics* 14, p. 306. DOI: 10.1186/1471-2105-14-306.

58. Karl, Stefan; Dandekar, Thomas (2015): Convergence behaviour and Control in Non-Linear Biological Networks. In *Scientific reports* 5, p. 9746. DOI: 10.1038/srep09746.

59. Kerkhofs, Johan; Roberts, Scott J.; Luyten, Frank P.; van Oosterwyck, Hans; Geris, Liesbet (2012): Relating the chondrocyte gene network to growth plate morphology: from genes to phenotype. In *PLoS ONE* 7 (4), pp. e34729. DOI: 10.1371/journal.pone.0034729.

60. Kiełbasa, Szymon M.; Vingron, Martin; Isalan, Mark (2008): Transcriptional Autoregulatory Loops Are Highly Conserved in Vertebrate Evolution. In *PLoS ONE* 3 (9), pp. e3210. DOI: 10.1371/journal.pone.0003210.

61. Kim, Junil; Vandamme, Drieke; Kim, Jeong-Rae; Munoz, Amaya Garcia; Kolch, Walter; Cho, Kwang-Hyun (2014): Robustness and evolvability of the human signaling network. In *PLoS computational biology* 10 (7), pp. e1003763.

62. Knuth, Donald Ervin (2011): Combinatorial algorithms. 1. pr. Upper Saddle River, N.J. [u.a.]: Addison-Wesley (The art of computer programming).

63. Koornneef, M.; Alonso-Blanco, C.; Blankestijn-de Vries, H.; Hanhart, C. J.; Peeters, A. J. (1998): Genetic interactions among late-flowering mutants of Arabidopsis. In *Genetics* 148 (2), pp. 885–892.

64. Koreth, John; Matsuoka, Ken-ichi; Kim, Haesook T.; McDonough, Sean M.; Bindra, Bhavjot; Alyea, Edwin P. et al. (2011): Interleukin-2 and regulatory T cells in graft-versus-host disease. In *The New England journal of medicine* 365 (22), pp. 2055–2066. DOI: 10.1056/NEJMoa1108188.

65. Kozomara, Ana; Griffiths-Jones, Sam (2014): miRBase: annotating high confidence microRNAs using deep sequencing data. In *Nucleic Acids Res.* 42 (Database issue), pp. D68-73. DOI: 10.1093/nar/gkt1181.

66. Kram, Brian W.; Xu, Wayne W.; Carter, Clay J. (2009): Uncovering the Arabidopsis thaliana nectary transcriptome: investigation of differential gene expression in floral nectariferous tissues. In *BMC Plant Biol.* 9, p. 92. DOI: 10.1186/1471-2229-9-92.

67. Kronenberg, Henry M. (2003): Developmental regulation of the growth plate. In *Nature* 423 (6937), pp. 332–336. DOI: 10.1038/nature01657.

68. Krumsiek, Jan; Pölsterl, Sebastian; Wittmann, Dominik M.; Theis, Fabian J. (2010): Odefy--from discrete to continuous models. In *BMC Bioinformatics* 11, p. 233. DOI: 10.1186/1471-2105-11-233.

69. Kwak, Hojoong; Lis, John T. (2013): Control of transcriptional elongation. In *Annual review of genetics* 47, pp. 483–508. DOI: 10.1146/annurev-genet-110711-155440.

70. Leblanc, Francis; Zhang, Dan; Liu, Xin; Loughran, Thomas P. (2012): Large granular lymphocyte leukemia: from dysregulated pathways to therapeutic targets. In *Future Oncol* 8 (7), pp. 787–801. DOI: 10.2217/fon.12.75.

71. Lecker, Stewart H.; Goldberg, Alfred L.; Mitch, William E. (2006): Protein degradation by the ubiquitin-proteasome pathway in normal and disease states. In *J. Am. Soc. Nephrol.* 17 (7), pp. 1807–1819. DOI: 10.1681/ASN.2006010083.

72. Leclerc, Robert D. (2008): Survival of the sparsest: robust gene networks are parsimonious. In *Mol Syst Biol* 4. DOI: 10.1038/msb.2008.52.

73. Lee, Tong Ihn; Young, Richard A. (2013): Transcriptional regulation and its misregulation in disease. In *Cell* 152 (6), pp. 1237–1251. DOI: 10.1016/j.cell.2013.02.014.

74. Lelli, Katherine M.; Slattery, Matthew; Mann, Richard S. (2012): Disentangling the many layers of eukaryotic transcriptional regulation. In *Annu. Rev. Genet.* 46, pp. 43–68. DOI: 10.1146/annurev-genet-110711-155437.

75. Liu, Yang-Yu; Slotine, Jean-Jacques; Barabási, Albert-László (2011): Controllability of complex networks. In *Nature* 473 (7346), pp. 167–173. DOI: 10.1038/nature10011.

76. Liu, Yang-Yu; Slotine, Jean-Jacques; Barabási, Albert-László (2012): Control centrality and hierarchical structure in complex networks. In *PLoS ONE* 7 (9), pp. e44459. DOI: 10.1371/journal.pone.0044459.

77. Liu, Z.; Tang, Y.; Qiu, T.; Cao, X.; Clemens, T. L. (2006): A Dishevelled-1/Smad1 Interaction Couples WNT and Bone Morphogenetic Protein Signaling Pathways in Uncommitted Bone Marrow Stromal Cells. In *Journal of Biological Chemistry* 281 (25), pp. 17156–17163. DOI: 10.1074/jbc.M513812200.

78. Lo, Kevin W-H; Ulery, Bret D.; Ashe, Keshia M.; Laurencin, Cato T. (2012): Studies of bone morphogenetic protein-based surgical repair. In *Advanced drug delivery reviews* 64 (12), pp. 1277–1291.

79. Luckheeram, Rishi Vishal; Zhou, Rui; Verma, Asha Devi; Xia, Bing (2012): $CD4^+$T cells: differentiation and functions. In *Clinical & developmental immunology* 2012, p. 925135. DOI: 10.1155/2012/925135.

80. Machado, Daniel; Costa, Rafael S.; Rocha, Miguel; Ferreira, Eugénio C.; Tidor, Bruce; Rocha, Isabel (2011): Modeling formalisms in Systems Biology. In *AMB Express* 1, p. 45. DOI: 10.1186/2191-0855-1-45.

81. Macneil, Lesley; Walhout, Albertha (2011): Gene regulatory networks and the role of robustness and stochasticity in the control of gene expression. In *Genome research* 21 (5), pp. 645–657. DOI: 10.1101/gr.097378.109.

82. Mar, Jessica C.; Quackenbush, John (2009): Decomposition of gene expression state space trajectories. In *PLoS computational biology* 5 (12), pp. e1000626. DOI: 10.1371/journal.pcbi.1000626.

83. Mendoza, Luis; Pardo, Fátima (2010): A robust model to describe the differentiation of T-helper cells. In *Theory Biosci.* 129 (4), pp. 283–293. DOI: 10.1007/s12064-010-0112-x.

84. Mendoza, Luis; Xenarios, Ioannis (2006): A method for the generation of standardized qualitative dynamical systems of regulatory networks. In *Theor Biol Med Model* 3, p. 13. DOI: 10.1186/1742-4682-3-13.

85. Minina, E.; Wenzel, H. M.; Kreschel, C.; Karp, S.; Gaffield, W.; McMahon, A. P.; Vortkamp, A. (2001): BMP and Ihh/PTHrP signaling interact to coordinate chondrocyte proliferation and differentiation. In *Development (Cambridge, England)* 128 (22), pp. 4523–4534.

86. Moon, Y.-H. (2003): EMF Genes Maintain Vegetative Development by Repressing the Flower Program in Arabidopsis. In *THE PLANT CELL ONLINE* 15 (3), pp. 681–693. DOI: 10.1105/tpc.007831.

87. Morris, Samantha A.; Cahan, Patrick; Li, Hu; Zhao, Anna M.; San Roman, Adrianna K; Shivdasani, Ramesh A. et al. (2014): Dissecting engineered cell types and enhancing cell fate conversion via CellNet. In *Cell* 158 (4), pp. 889–902. DOI: 10.1016/j.cell.2014.07.021.

88. Muller, H. J. (1932): Further Studies on the Nature and Causes of Gene Mutations. In *Proceedings of the Sixth International Congress of Genetics* I, pp. 213–255.

89. Müller, Franz-Josef; Loring, Jeanne F. (2014): Network biology: A compass for stem-cell differentiation. In *Nature* 513 (7519), pp. 498–499. DOI: 10.1038/513498a.

90. Müller, Franz-Josef; Schuppert, Andreas (2011): Few inputs can reprogram biological networks. In *Nature* 478 (7369), pp. E4. DOI: 10.1038/nature10543.

91. Nacher, Jose C.; Akutsu, Tatsuya (2013): Structural controllability of unidirectional bipartite networks. In *Sci. Rep.* 3. DOI: 10.1038/srep01647.

92. Nagano, Takashi; Fraser, Peter (2011): No-nonsense functions for long noncoding RNAs. In *Cell* 145 (2), pp. 178–181. DOI: 10.1016/j.cell.2011.03.014.

93. Naseem, Muhammad; Philippi, Nicole; Hussain, Anwar; Wangorsch, Gaby; Ahmed, Nazeer; Dandekar, Thomas (2012): Integrated systems view on networking by hormones in Arabidopsis immunity reveals multiple crosstalk for cytokinin. In *Plant Cell* 24 (5), pp. 1793–1814. DOI: 10.1105/tpc.112.098335.

94. Ng, Huck-Hui; Surani, M. Azim (2011): The transcriptional and signalling networks of pluripotency. In *Nat. Cell Biol.* 13 (5), pp. 490–496. DOI: 10.1038/ncb0511-490.

95. Nicosia, V.; Criado, R.; Romance, M.; Russo, G.; Latora, V. (2012): Controlling centrality in complex networks. In *Scientific reports* 2, p. 218. DOI: 10.1038/srep00218.

96. Niederst, Matthew J.; Engelman, Jeffrey A. (2013): Bypass mechanisms of resistance to receptor tyrosine kinase inhibition in lung cancer. In *Science signaling* 6 (294), pp. re6.

97. Nurieva, Roza I.; Chung, Yeonseok; Hwang, Daehee; Yang, Xuexian O.; Kang, Hong Soon; Ma, Li et al. (2008): Generation of T follicular helper cells is mediated by interleukin-21 but independent of T helper 1, 2, or 17 cell lineages. In *Immunity* 29 (1), pp. 138–149. DOI: 10.1016/j.immuni.2008.05.009.

98. Omori, Kenji; Kotera, Jun (2007): Overview of PDEs and their regulation. In *Circ. Res.* 100 (3), pp. 309–327. DOI: 10.1161/01.RES.0000256354.95791.f1.

99. Orkin, Stuart H.; Hochedlinger, Konrad (2011): Chromatin connections to pluripotency and cellular reprogramming. In *Cell* 145 (6), pp. 835–850. DOI: 10.1016/j.cell.2011.05.019.

100. Page, Lawrence; Brin, Sergey; Motwani, Rajeev; Winograd, Terry (1999): The PageRank Citation Ranking: Bringing Order to the Web. Stanford InfoLab. Available online at http://ilpubs.stanford.edu:8090/422/.

101. Pan, Yujian; Li, Xiang (2014): Structural controllability and controlling centrality of temporal networks. In *PLoS ONE* 9 (4), pp. e94998. DOI: 10.1371/journal.pone.0094998.

102. Panne, Daniel (2008): The enhanceosome. In *Current opinion in structural biology* 18 (2), pp. 236–242. DOI: 10.1016/j.sbi.2007.12.002.

103. Paz, Arnon; Brownstein, Zippora; Ber, Yaara; Bialik, Shani; David, Eyal; Sagir, Dorit et al. (2011): SPIKE: a database of highly curated human signaling pathways. In *Nucleic Acids Res.* 39 (Database issue), pp. D793-9. DOI: 10.1093/nar/gkq1167.

104. Piróg, Katarzyna A.; Briggs, Michael D. (2010): Skeletal dysplasias associated with mild myopathy-a clinical and molecular review. In *Journal of biomedicine & biotechnology* 2010, p. 686457. DOI: 10.1155/2010/686457.

105. Poole, Anthony M.; Phillips, Matthew J.; Penny, David (2003): Prokaryote and eukaryote evolvability. In *Biosystems* 69 (2-3), pp. 163–185. DOI: 10.1016/S0303-2647(02)00131-4.

106. Pu, Cun-Lai; Pei, Wen-Jiang; Michaelson, Andrew (2012): Robustness analysis of network controllability. In *Physica A: Statistical Mechanics and its Applications* 391 (18), pp. 4420–4425. DOI: 10.1016/j.physa.2012.04.019.

107. Pufall, Miles A.; Graves, Barbara J. (2002): Autoinhibitory domains: modular effectors of cellular regulation. In *Annual review of cell and developmental biology* 18, pp. 421–462. DOI: 10.1146/annurev.cellbio.18.031502.133614.

108. Qian, Xiaoning; Dougherty, Edward R. (2013): Validation of gene regulatory network inference based on controllability. In *Frontiers in genetics* 4, p. 272. DOI: 10.3389/fgene.2013.00272.

109. Ritter, Stefanie L.; Hall, Randy A. (2009): Fine-tuning of GPCR activity by receptor-interacting proteins. In *Nat. Rev. Mol. Cell Biol.* 10 (12), pp. 819–830. DOI: 10.1038/nrm2803.

110. Robert-Seilaniantz, Alexandre; Grant, Murray; Jones, Jonathan D G (2011): Hormone crosstalk in plant disease and defense: more than just jasmonate-salicylate antagonism. In *Annual review of phytopathology* 49, pp. 317–343. DOI: 10.1146/annurev-phyto-073009-114447.

111. Rorsman, Patrik; Braun, Matthias (2013): Regulation of insulin secretion in human pancreatic islets. In *Annual review of physiology* 75, pp. 155–179. DOI: 10.1146/annurev-physiol-030212-183754.

112. Ruths, Justin; Ruths, Derek (2014a): Control profiles of complex networks. In *Science (New York, N.Y.)* 343 (6177), pp. 1373–1376. DOI: 10.1126/science.1242063.

113. Ruths, Justin; Ruths, Derek (2014b): Network models. Response to Comment on "Control profiles of complex networks". In *Science (New York, N.Y.)* 346 (6209), p. 561. DOI: 10.1126/science.1256714.

114. Saadoun, David; Rosenzwajg, Michelle; Joly, Florence; Six, Adrien; Carrat, Fabrice; Thibault, Vincent et al. (2011): Regulatory T-cell responses to low-dose interleukin-2 in HCV-induced vasculitis. In *The New England journal of medicine* 365 (22), pp. 2067–2077. DOI: 10.1056/NEJMoa1105143.

115. Salgado, Heladia; Peralta-Gil, Martin; Gama-Castro, Socorro; Santos-Zavaleta, Alberto; Muñiz-Rascado, Luis; García-Sotelo, Jair S. et al. (2013): RegulonDB v8.0: omics data sets, evolutionary conservation, regulatory phrases, cross-validated gold standards and more. In *Nucleic Acids Res.* 41 (Database issue), pp. D203-13. DOI: 10.1093/nar/gks1201.

116. Sánchez-Corrales, Yara-Elena; Alvarez-Buylla, Elena R.; Mendoza, Luis (2010): The Arabidopsis thaliana flower organ specification gene regulatory network determines a robust differentiation process. In *J. Theor. Biol.* 264 (3), pp. 971–983. DOI: 10.1016/j.jtbi.2010.03.006.

117. Sánchez-Corrales, Yara-Elena; Álvarez-Buylla, Elena R.; Mendoza, Luis (2011): Corrigendum to "The Arabidopsis thaliana flower organ specification gene regulatory network determines a robust differentiation process" [J. Theor. Biol. 264 (2010) 971–983]. In *Journal of Theoretical Biology* 274 (1), p. 185. DOI: 10.1016/j.jtbi.2011.01.027.

118. Shannon, S.; Meeks-Wagner, D. R. (1993): Genetic Interactions That Regulate Inflorescence Development in Arabidopsis. In *Plant Cell* 5 (6), pp. 639–655. DOI: 10.1105/tpc.5.6.639.

119. Solomon, Lauren A.; Bérubé, Nathalie G.; Beier, Frank (2008): Transcriptional regulators of chondrocyte hypertrophy. In *Birth defects research. Part C, Embryo today : reviews* 84 (2), pp. 123–130. DOI: 10.1002/bdrc.20124.

120. Steinert, Andre F.; Weissenberger, Manuel; Kunz, Manuela; Gilbert, Fabian; Ghivizzani, Steven C.; Göbel, Sascha et al. (2012): Indian hedgehog gene transfer is a chondrogenic inducer of human mesenchymal stem cells. In *Arthritis Res Ther* 14 (4), pp. R168. DOI: 10.1186/ar3921.

121. Tang, Xiaodong; Lu, Xinchang; Guo, Wei; Ren, Tingting; Zhao, Hui; Zhao, Fulong; Tang, Guoqing (2010): Different expression of Sox9 and Runx2 between chondrosarcoma and dedifferentiated chondrosarcoma cell line. In *European Journal of Cancer Prevention* 19 (6), pp. 466–471. DOI: 10.1097/CEJ.0b013e32833d942f.

122. Taylor, S. S.; Yang, J.; Wu, J.; Haste, N. M.; Radzio-Andzelm, E.; Anand, G. (2004): PKA: a portrait of protein kinase dynamics. In *Biochim. Biophys. Acta* 1697 (1-2), pp. 259–269. DOI: 10.1016/j.bbapap.2003.11.029.

123. Teixeira, Miguel Cacho; Monteiro, Pedro Tiago; Guerreiro, Joana Fernandes; Gonçalves, Joana Pinho; Mira, Nuno Pereira; dos Santos, Sandra Costa et al. (2014): The YEASTRACT database: an upgraded information system for the analysis of gene and genomic transcription regulation in Saccharomyces cerevisiae. In *Nucleic Acids Res.* 42 (Database issue), pp. D161-6. DOI: 10.1093/nar/gkt1015.

124. Vaquerizas, Juan M.; Kummerfeld, Sarah K.; Teichmann, Sarah A.; Luscombe, Nicholas M. (2009): A census of human transcription factors: function, expression and evolution. In *Nat. Rev. Genet.* 10 (4), pp. 252–263. DOI: 10.1038/nrg2538.

125. Veldhoen, Marc; Uyttenhove, Catherine; van Snick, Jacques; Helmby, Helena; Westendorf, Astrid; Buer, Jan et al. (2008): Transforming growth factor-beta 'reprograms' the differentiation of T helper 2 cells and promotes an interleukin 9-producing subset. In *Nature immunology* 9 (12), pp. 1341–1346. DOI: 10.1038/ni.1659.

126. Wagner, Andreas (2005): Distributed robustness versus redundancy as causes of mutational robustness. In *BioEssays : news and reviews in molecular, cellular and developmental biology* 27 (2), pp. 176–188. DOI: 10.1002/bies.20170.

127. Wagner, Klaus W. (2003): Theoretische Informatik. Eine kompakte Einführung. 2., überarb. Aufl. Berlin, Heidelberg, New York, Hongkong, London, Mailand, Paris, Tokio: Springer (Springer-Lehrbuch).

128. Wehrli, Bret Michael; Huang, Wendong; Crombrugghe, Benoit de; Ayala, Alberto G.; Czerniak, Bogdan (2003): Sox9, a master regulator of chondrogenesis, distinguishes mesenchymal chondrosarcoma from other small blue round cell tumors. In *Hum. Pathol.* 34 (3), pp. 263–269. DOI: 10.1053/hupa.2003.41.

129. Weinstein, Nathan; Mendoza, Luis (2012): Building Qualitative Models of Plant Regulatory Networks with SQUAD. In *Frontiers in plant science* 3, p. 72. DOI: 10.3389/fpls.2012.00072.

130. Whaley, John (2007): JavaBDD. Available online at http://javabdd.sourceforge.net/, checked on 10/30/2014.

131. Wittmann, Dominik M.; Krumsiek, Jan; Saez-Rodriguez, Julio; Lauffenburger, Douglas A.; Klamt, Steffen; Theis, Fabian J. (2009): Transforming Boolean models to continuous models: methodology and application to T-cell receptor signaling. In *BMC Syst Biol* 3, p. 98. DOI: 10.1186/1752-0509-3-98.

132. Xing, Junchao; Wu, Yuzhang; Ni, Bing (2011): Th9: a new player in asthma pathogenesis? In *The Journal of asthma : official journal of the Association for the Care of Asthma* 48 (2), pp. 115–125. DOI: 10.3109/02770903.2011.554944.

133. Yao, Xin; Huang, Jiaqi; Zhong, Haihong; Shen, Nan; Faggioni, Raffaella; Fung, Michael; Yao, Yihong (2014): Targeting interleukin-6 in inflammatory autoimmune diseases and cancers. In *Pharmacology & therapeutics* 141 (2), pp. 125–139. DOI: 10.1016/j.pharmthera.2013.09.004.

134. Young, Richard A. (2011): Control of the embryonic stem cell state. In *Cell* 144 (6), pp. 940–954. DOI: 10.1016/j.cell.2011.01.032.

135. Yu, Aixi; Hou, Ming; Xie, Xiaoqing; Li, Peng; Zhu, Zhiqi (2014): Effects of Sox9 gene therapy on the healing of bone-tendon junction: An experimental study. In *Indian J Orthop* 48 (1), p. 88. DOI: 10.4103/0019-5413.125521.

136. yWorks GmbH (2014): yEd Graph Editor. Available online at http://www.yworks.com/en/products/yfiles/yed/, checked on 10/30/2014.

137. Zambrano-Zaragoza, José Francisco; Romo-Martínez, Enrique Jhonatan; Durán-Avelar, Ma de Jesús; García-Magallanes, Noemí; Vibanco-Pérez, Norberto (2014): Th17 cells in autoimmune and infectious diseases. In *International journal of inflammation* 2014, p. 651503. DOI: 10.1155/2014/651503.

138. Zhang, Ranran; Shah, Mithun Vinod; Yang, Jun; Nyland, Susan B.; Liu, Xin; Yun, Jong K. et al. (2008): Network model of survival signaling in large granular lymphocyte leukemia. In *Proc. Natl. Acad. Sci. U.S.A.* 105 (42), pp. 16308–16313. DOI: 10.1073/pnas.0806447105.

139. Zhang, S-J; Wang, L.; Ming, L.; Guo, X-B; Wang, H-M; Li, X-W; Li, L. (2013): Blockade of IL-6 signal exacerbates acute inflammatory bowel disease via inhibiting

IL-17 producing in activated CD4+ Th17 population. In *Eur Rev Med Pharmacol Sci* 17 (24), pp. 3291–3295.

140. Zhu, Jinfang; Paul, William E. (2008): CD4 T cells: fates, functions, and faults. In *Blood* 112 (5), pp. 1557–1569. DOI: 10.1182/blood-2008-05-078154.

141. Zhu, Jinfang; Yamane, Hidehiro; Paul, William E. (2010): Differentiation of effector CD4 T cell populations (*). In *Annual review of immunology* 28, pp. 445–489. DOI: 10.1146/annurev-immunol-030409-101212.

6 Previously published material

The following statements are in accordance with guidelines kindly provided by Prof. Thomas Hünig as head of the Doctoral Committee (Promotionskommission) of the Faculty of Medicine.

Several parts of this thesis have been published already. Every previously published part also appearing in this thesis has been written solely by Stefan Karl.

6.1 List of all sections

Section 1.3.1: Parts of the text are adapted from the (untitled) introductory section in (Karl, Dandekar 2015).

Section 1.3.3: Parts of the text are adapted from the introductory section in (Karl, Dandekar 2015)

Section 1.3.7: Parts of the text are adapted from the “Background” section in (Karl, Dandekar 2013).

Section 1.4: Parts of the text are adapted from the “Obtaining the stable steady states for discrete models from the Boolean tree” section in (Karl, Dandekar 2013).

Section 1.5: Parts of the text are adapted from the “Results” section in (Karl, Dandekar 2015).

Section 1.7: Parts of the text are adapted from the “Introduction” section in (Karl, Dandekar 2015).

Section 2.2.1: Parts of the text are adapted from the sections “A recursive algorithm to calculate the BooleCube polynomial” and “Speed up of the BooleCube calculation” in (Karl, Dandekar 2013).

Section 2.2.2: Parts of the text are adapted from the section “Obtaining the stable steady states for discrete models from the Boolean tree” in (Karl, Dandekar 2013).

Section 2.3: Parts of the text are adapted from the Supplementary Information to (Karl, Dandekar 2015), contents of which are also included in the main text of (Karl, Dandekar 2015).

Section 3.1.1: Parts of the text are adapted from the section “Speed up of the BooleCube calculation” in (Karl, Dandekar 2013).

Section 3.1.2: Parts of the text are adapted from the section “Speed of the SSS calculation” and the “Additional File 1” in (Karl, Dandekar 2013).

Section 3.2: Parts of the text are adapted from the “Results” section in (Karl, Dandekar 2015).

Section 3.2.1: Parts of the text are adapted from the section 1.3 in the Supplementary Information to (Karl, Dandekar 2015).

Section 3.3: Parts of the text are adapted from the section “Controllability in biological networks” in (Karl, Dandekar 2015).

Section 3.3.1: Parts of the text are adapted from the Supplementary Information for (Karl, Dandekar 2015) and from the section “Applied example: Arabidopsis thaliana development” in (Karl, Dandekar 2013).

Section 3.3.2: Parts of the text are adapted from the section “Applied example: Arabidopsis thaliana development” in (Karl, Dandekar 2013).

Section 3.3.6: Parts of the text are adapted from the section “Applied example II: Arabidopsis thaliana immunity and pathogen Pst DC3000” in (Karl, Dandekar 2013).

Section 4: Parts of the text are adapted from the “Discussion” section in (Karl, Dandekar 2015).

Section 5: Parts of the text are adapted from the abstract of (Karl, Dandekar 2015).

Section 7.3: The text is adapted from the Additional File 1 in (Karl, Dandekar 2013).

Section 7.4: The data is taken from the Supplementary Tables 1-10 and the Supplementary Figures 1-5 from (Karl, Dandekar 2015).

Section 7.5: The data is included in Supplementary Information 2 in (Karl, Dandekar 2015).

Figure 2: Parts c-g of the figure are identical to Figure 1 in (Karl, Dandekar 2013).

Figure 3: Parts of the figure have been published in Jimena’s tutorials on http://stefan-karl.de/jimena

Figure 5: The figure is similar to Figure 2 in (Karl, Dandekar 2013).

Figure 6: The figure is similar to Figure 3 in (Karl, Dandekar 2013).

Figure 7: The figure has been adapted from Figure 1a-f in (Karl, Dandekar 2015).

Figure 8: The figure is identical to Figure 1 g,h in (Karl, Dandekar 2015), and the caption contains text from the “Methods” sections of (Karl, Dandekar 2015).

Figure 9: The caption contains text from the “Methods” sections of (Karl, Dandekar 2015).

Figure 14: The parts of the figure are identical to Figure 4 and 5 in (Karl, Dandekar 2013).

Figure 15: The figure is identical to Figure 6 in (Karl, Dandekar 2013).

Figure 16: The parts of the figure are identical to Figure 2 in (Karl, Dandekar 2015).

Figure 18: See Copyrighted figures.

Figure 19: The figure has been published previously on the Jimena website.

Figure 20: The parts of the figure are identical to Figure 3a,c and Figure 4a in (Karl, Dandekar 2015).

Figure 21: The figure is identical to Figure 3b in (Karl, Dandekar 2015).

Figure 22: The figure is identical to Figure 3d in (Karl, Dandekar 2015).

Figure 25: See Copyrighted figures.

Figure 26: The figure is identical to Figure 4b in (Karl, Dandekar 2015).

Figure 27: The figure is identical to the Supplementary Figure 3 in (Karl, Dandekar 2015).

Table 1: Part b is similar to Table 2 in (Karl, Dandekar 2013), although the data has been recalculated.

Table 2: The table is identical to the "Supplementary Table 1" in (Karl, Dandekar 2015).

Table 3: Parts of the table are taken from Table 1 in (Karl, Dandekar 2015).

6.2 Copyright

6.2.1 (Karl, Dandekar 2013)

6.2.2 (Karl, Dandekar 2015)

6.2.3 Jimena

The Jimena simulation package, its source code and the graphical user interface were published under the GNU Lesser General Public License 3.0 (GNU LGPL) (https://www.gnu.org/licenses/lgpl-3.0.en.html) and are available at http://stefan-karl.de/jimena/ and http://bioinfo.biozentrum.uni-wuerzburg.de/computing/jimena-c/.

6.2.4 Copyrighted figures

Figure 18: Adapted from (Kram et al. 2009), Figure 1A. The figure was distributed under the Creative Commons Attribution License 2.0 which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. (http://creativecommons.org/licenses/by/2.0/)

Figure 25: Adapted from (Piróg, Briggs 2010), Figure 2. The figure was distributed under the Creative Commons Attribution License 3.0 which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. (http://creativecommons.org/licenses/by/3.0/)

7 Appendix

7.1 Generation of random networks

In Jimena's random network generator, Erdős-Rényi networks are created by randomly distributing the desired number of edges over the nodes and discarding the result if the network is not fully connected. Scale-free networks are grown to the desired size by preferential attachment according to the Barabási-Albert model (Albert 2002): Starting from a seed of 2 nodes connected by either an activating or an inhibiting connection, new nodes are iteratively added to the existing network. The number of connections each of these new nodes has to the existing network is chosen in accordance with the desired number of connections in the generated network. The type of the interactions is chosen

randomly between inhibiting (50%) and activating (50%) influences. If specific network parameters such as the number of input nodes are desired, random networks fulfilling these criteria are chosen randomly from the generated networks.

Simplified versions of the algorithms are given in pseudocode below.

7.1.1 Random Erdős–Rényi networks

```
function randomERNetwork (numberOfNodes, numberOfConnections, activatingRatio)
    nodes = {x1,...,xnumberOfNodes} // Nodes of the network
    connections = {} // Connection of the network
    do numberOfConnections times
        sourceNode = getRandomElement(nodes) // Get a random source node
        target Node = getRandomElement(nodes) // Get a random target node
        activating = randomBoolean( activatingRatio)
        if not connections.contains(sourceNode, targetNode, activating) then
            // Do not allow duplicate connections
            connections.add(sourceNode, targetNode, activating)
        else
            numberOfConnections = numberOfConnections + 1
        end if
    end do
    result = constructNetwork(nodes, connections) // Construct a network
    if not isConnected(result) then
        // If the network is not connected, try again
        return randomERNetwork (numberOfNodes, numberOfConnections, activatingRatio)
    end if
    return result
end function

function randomBoolean (activatingRatio)
    // Returns true in activatingRatio * 100% of all cases
    if random() < activatingRatio then
        // random() returns a random value between 0 and 1
        return true
    else
        return false
    end if
end function
```

7.1.2 Random scale-free networks

```
function randomSFNetwork (numberOfNodes, numberOfConnections, activatingRatio)
    nodes = {x1,...,xnumberOfNodes} // Nodes of the network
    connections = {} // Connection of the network
    connections.add(x1, x2, randomBoolean(activatingRatio)) // Add a seed connection
    for all node in nodes do
        repetitions = roundUp(numberOfConnections/numberOfNodes)
        do repetitions times
```

```
            // Construct enough temporary connections
            if randomBoolean(0.5) then
                otherNode = getSource(getRandomElement(connections))
            else
                otherNode = getTarget(getRandomElement(connections))
            end if
            if randomBoolean(0.5) then
                targetNode = otherNode
                sourceNode = node
            else
                targetNode = node
                sourceNode = otherNode
            end if
            if not connections.contains(sourceNode, targetNode, activating) then
                // Do not allow duplicate connections
                connections.add(sourceNode, targetNode, activating)
            else
                repetitions = repetitions + 1
            end if
        end do
    end for
    // Remove superfluous nodes
    while connections.size > numberOfConnections
        // Removing random connections preserves the distribution
        connections.delete(getRandomElement(connections))
    end while
    result = constructNetwork(nodes, connections) // Construct a network
    if not isConnected(result) then
        // If the network is not connected, try again
        return randomSFNetwork (numberOfNodes, numberOfConnections, activatingRatio)
    end if
    return result
end function
```

7.2 Jimena's multithreading strategy

Each calculation is divided into several thousand subcalculations encapsulated in a java.lang.concurrrent.Future<?> class, representing for example one run in the approximation of the control centralities. The Futures are then fed into the custom built executor class JimenaExecutor, which extends a standard java.util.concurrent.ThreadPoolExecutor with error handling functions. The executor dynamically distributes the Futures on the computational cores based on their availability. Once all calculations in the executor have finished, the subresults are extracted and combined to the final result by the multithreading algorithm.

7.3 Proof summary of Jimena's interpolation algorithm

This proof demonstrates that the interpolation function $C[B]$ of a Boolean function $B(...)$ with tree nodes n_i in a network with nodes x_i can be assembled stepwise from the leaves (= inputs) of the tree using the algorithm from section 2.2.1. First, consider input nodes, i.e. nodes n_k for which $f_k(x_j) = x_j$ for some x_j. $C[f_k]$ can be determined by simply applying the Odefy polynomial:

$$C\left[f_k(x_j)\right] = \sum_{\bar{x}_j=0}^{1} f_k(\bar{x}_j) \prod_{i=j}^{j} \xi(x_i, \bar{x}_i) = \sum_{\bar{x}_j=0}^{1} f_k(\bar{x}_j)\,\xi(x_j, \bar{x}_j) = f_k(x_j) = x_j$$

The second case in the algorithm are negating nodes, the only unary nodes in the tree. For a negating node n_k whose input node n_i has a function $f_i(x_1,...,x_n)$ it must be shown that $C[f_k] = C[\neg f_i] = 1 - C[f_i]$:

$$1 - C[f_i] = 1 - \sum_{\bar{x}_1=0}^{1} \cdot ... \cdot \sum_{\bar{x}_n=0}^{1} f_i(\bar{x}_1,...,\bar{x}_n) \cdot \prod_{i=1}^{n} \xi(x_i, \bar{x}_i)$$

By setting

$$1 = \sum_{\bar{x}_1=0}^{1} \cdot ... \cdot \sum_{\bar{x}_n=0}^{1} 1 \cdot \prod_{i=1}^{n} \xi(x_i, \bar{x}_i)$$

which can be easily proven by induction over n one gets

$$= \sum_{\bar{x}_1=0}^{1} \cdot ... \cdot \sum_{\bar{x}_n=0}^{1} 1 \cdot \prod_{i=1}^{n} \xi(x_i, \bar{x}_i) - \sum_{\bar{x}_1=0}^{1} \cdot ... \cdot \sum_{\bar{x}_n=0}^{1} f_i(\bar{x}_1,\ldots,\bar{x}_n) \cdot \prod_{i=1}^{n} \xi(x_i, \bar{x}_i)$$

$$= \sum_{\bar{x}_1=0}^{1} \cdot ... \cdot \sum_{\bar{x}_n=0}^{1} \left(1 - f_i(\bar{x}_1,\ldots,\bar{x}_n)\right) \cdot \prod_{i=1}^{n} \xi(x_i, \bar{x}_i)$$

$$= \sum_{\bar{x}_1=0}^{1} \cdot ... \cdot \sum_{\bar{x}_n=0}^{1} (\neg f)_i(\bar{x}_1,...,\bar{x}_n) \cdot \prod_{i=1}^{n} \xi(x_i, \bar{x}_i) = C[\neg f_i]$$

The last case in the algorithm considers a binary node with two inputs $f_{i_1}\left(x_{1,1},\ldots,x_{1,n_1}\right)$ and $f_{i_2}\left(x_{2,1},\ldots,x_{2,n_2}\right)$ which represents a logic gate $\otimes:\{0,1\}^2\to\{0,1\}$. One can now show that $C\left[f_{i_1}\otimes f_{i_2}\right]=C[\otimes](C\left[f_{i_1}\right],C\left[f_{i_1}\right])$:

$$C[\otimes]\left(C\left[f_{i_1}\right],C\left[f_{i_2}\right]\right)=$$

$$\sum_{\bar{a}_1=0}^{1}\sum_{\bar{a}_2=0}^{1}(\bar{a}_1\otimes\bar{a}_2)\cdot\prod_{\varphi=1}^{2}\begin{cases}\sum_{\bar{x}_{\varphi,1}=0}^{1}\cdot\ldots\cdot\sum_{\bar{x}_{\varphi,n_\varphi}=0}^{1}f_{i_\varphi}\left(x_{\varphi,1},\ldots,x_{\varphi,n_\varphi}\right)\prod_{i=1}^{n_\varphi}\xi\left(x_{\varphi,i},\bar{x}_{\varphi,i}\right) & :a_\varphi=1\\ 1-\sum_{\bar{x}_{\varphi,1}=0}^{1}\cdot\ldots\cdot\sum_{\bar{x}_{\varphi,n_\varphi}=0}^{1}f_{i_\varphi}\left(x_{\varphi,1},\ldots,x_{\varphi,n_\varphi}\right)\prod_{i=1}^{n_\varphi}\xi\left(x_{\varphi,i},\bar{x}_{\varphi,i}\right) & :a_\varphi=0\end{cases}$$

Using the proof for negating nodes yields

$$=\sum_{\bar{a}_1=0}^{1}\sum_{\bar{a}_2=0}^{1}(\bar{a}_1\otimes\bar{a}_2)\cdot\prod_{\varphi=1}^{2}\begin{cases}\sum_{\bar{x}_{\varphi,1}=0}^{1}\cdot\ldots\cdot\sum_{\bar{x}_{\varphi,n_\varphi}=0}^{1}f_{i_\varphi}\left(x_{\varphi,1},\ldots,x_{\varphi,n_\varphi}\right)\prod_{i=1}^{n_\varphi}\xi\left(x_{\varphi,i},\bar{x}_{\varphi,i}\right) & :a_\varphi=1\\ \sum_{\bar{x}_{\varphi,1}=0}^{1}\cdot\ldots\cdot\sum_{\bar{x}_{\varphi,n_\varphi}=0}^{1}\left(\neg f_{i_\varphi}\right)\left(x_{\varphi,1},\ldots,x_{\varphi,n_\varphi}\right)\prod_{i=1}^{n_\varphi}\xi\left(x_{\varphi,i},\bar{x}_{\varphi,i}\right) & :a_\varphi=0\end{cases}$$

and by expanding the product and pulling in the $(\bar{a}_1\otimes\bar{a}_2)$

$$=\sum_{\bar{a}_1=0}^{1}\sum_{\bar{a}_2=0}^{1}\sum_{\bar{x}_{1,1}=0}^{1}\cdot\ldots\cdot\sum_{\bar{x}_{2,n_2}=0}^{1}(\bar{a}_1\otimes\bar{a}_2)\cdot\prod_{\varphi=1}^{2}\left(\prod_{i=1}^{n_\varphi}\xi\left(x_{\varphi,i},\bar{x}_{\varphi,i}\right)\begin{cases}f_{i_\varphi}\left(x_{\varphi,1},\ldots,x_{\varphi,n_\varphi}\right) & :a_\varphi=1\\ \left(\neg f_{i_\varphi}\right)\left(x_{\varphi,1},\ldots,x_{\varphi,n_\varphi}\right) & :a_\varphi=0\end{cases}\right)$$

$$=\sum_{\bar{x}_{1,1}=0}^{1}\cdot\ldots\cdot\sum_{\bar{x}_{2,n_2}=0}^{1}(f_{i_1}\otimes f_{i_2})\left(x_{1,1},\ldots x_{2,n_2}\right)\prod_{i=1}^{n_1}\xi\left(x_{1,i},\bar{x}_{1,i}\right)\prod_{i=1}^{n_2}\xi\left(x_{2,i},\bar{x}_{2,i}\right)$$

$$=C[f_1\otimes f_2]$$

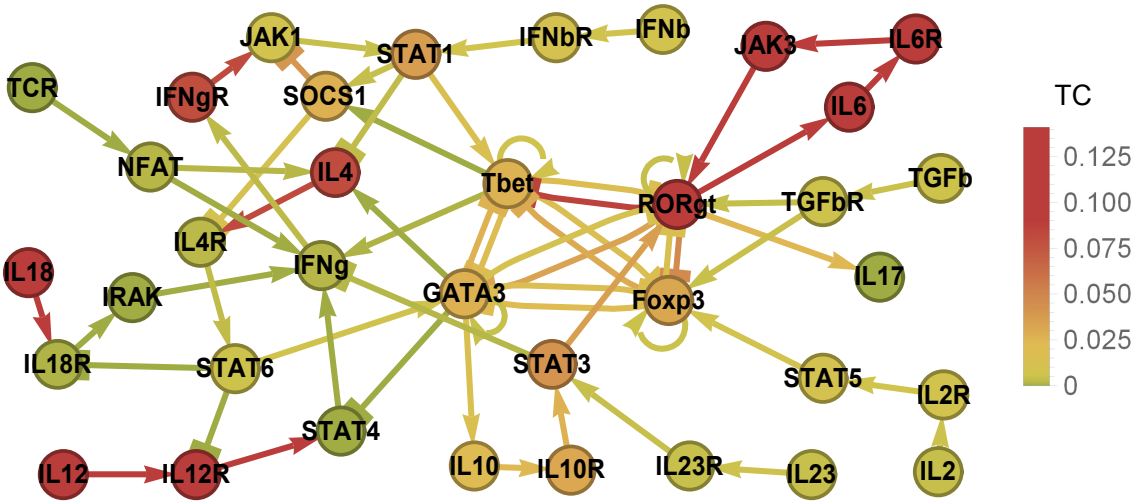
The algorithm can therefore stepwise assemble or calculate the interpolation function by using $C[\neg f]=1-C[f]$ and $C[f_1\otimes f_2]=C[\otimes](C[f_1],C[f_2])$.

7.4 Centrality in biological networks

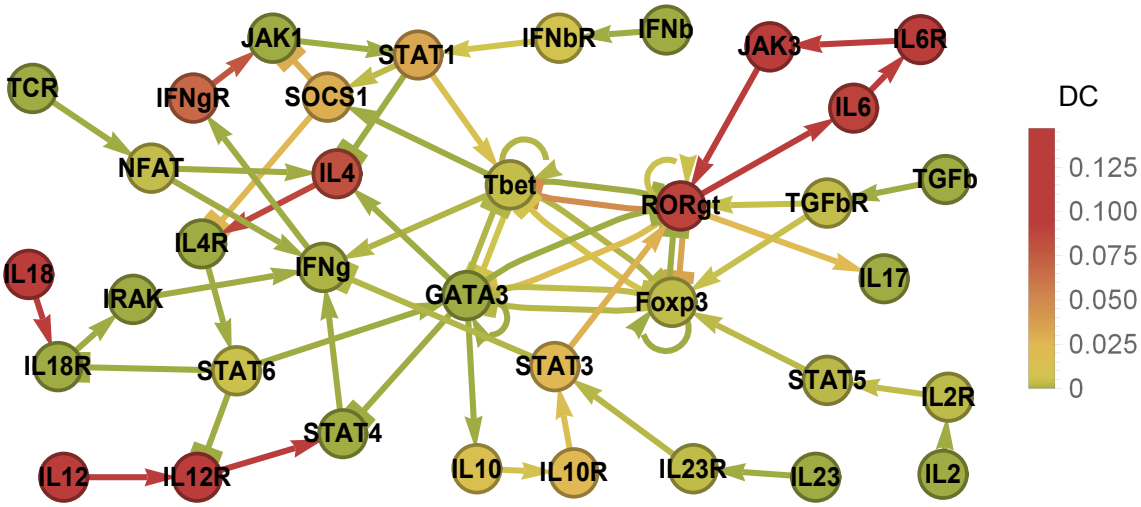
This chapter provides additional data on the networks in this thesis.

7.4.1 T-helper differentiation (without input loops)

7.4.1.1 Total centrality



7.4.1.2 Dynamic centrality



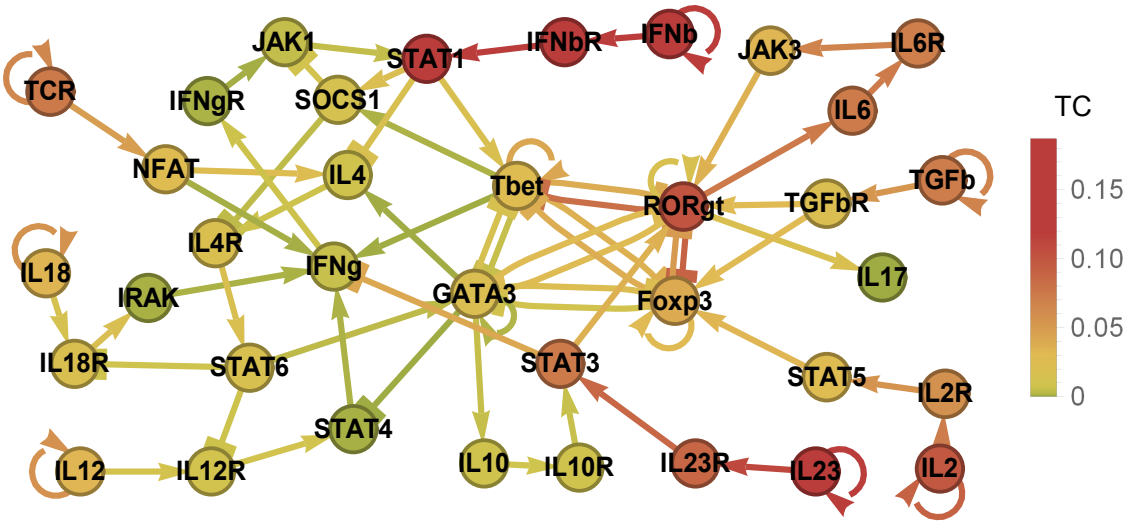
7.4.1.3 Additional data

Node	TC	VC	DC	TC-Sens.	VC-Sens.	Outgoing	Incoming	Loop
Foxp3	3.1E-02	1.3E-02	2.7E-03	3.1E-04	1.4E-16	4	6	yes
GATA3	2.7E-02	2.0E-02	3.0E-16	1.3E-02	1.2E-03	7	5	yes
IFNb	6.6E-03	7.1E-03	1.2E-16	1.2E-19	3.5E-20	1	0	no
IFNbR	9.5E-03	2.2E-03	6.7E-03	3.6E-18	1.0E-18	1	1	no
IFNg	1.3E-03	6.4E-04	5.2E-04	8.8E-02	3.1E-03	1	5	no
IFNgR	8.0E-02	5.1E-04	6.8E-02	1.1E-01	5.7E-03	1	1	no
IL10	2.1E-02	4.1E-03	1.4E-02	1.3E-02	1.2E-03	1	1	no
IL10R	3.1E-02	7.2E-04	2.4E-02	1.3E-02	1.2E-03	1	1	no
IL12	1.4E-01	5.3E-19	1.3E-01	1.3E-19	4.6E-20	1	0	no

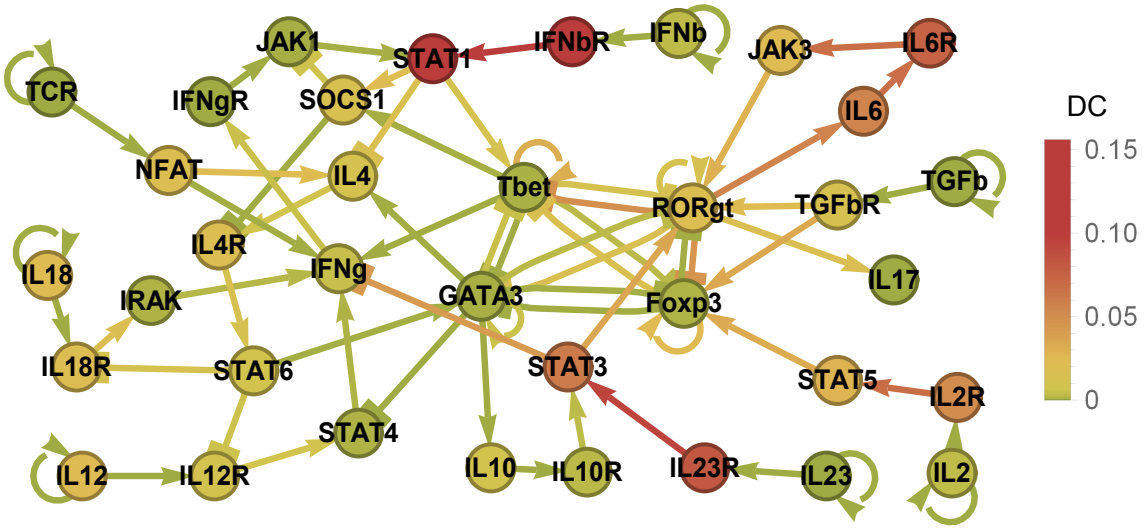
IL12R	1.0E-01	6.8E-19	1.1E-01	2.7E-02	6.5E-19	1	2	no
IL17	0.0E+00	4.0E-21	0.0E+00	1.4E-01	2.6E-02	0	1	no
IL18	1.4E-01	1.3E-18	1.4E-01	1.4E-19	4.2E-20	1	0	no
IL18R	6.4E-04	1.4E-18	5.9E-18	2.7E-02	5.6E-19	1	2	no
IL2	3.7E-03	4.8E-03	1.6E-16	1.2E-19	3.3E-20	1	0	no
IL23	3.9E-03	2.8E-03	1.7E-17	1.4E-19	4.3E-20	1	0	no
IL23R	3.9E-03	1.5E-03	2.7E-03	4.3E-18	1.3E-18	1	1	no
IL2R	6.7E-03	4.5E-03	2.2E-03	3.6E-18	9.6E-19	1	1	no
IL4	8.1E-02	2.4E-17	8.1E-02	1.1E-02	1.1E-03	1	3	no
IL4R	1.8E-03	2.2E-03	8.6E-17	3.5E-02	1.2E-03	1	2	no
IL6	1.2E-01	1.1E-02	9.1E-02	1.4E-01	2.6E-02	1	1	no
IL6R	1.2E-01	3.3E-03	1.1E-01	1.4E-01	2.6E-02	1	1	no
IRAK	3.8E-06	3.1E-19	1.6E-06	2.7E-02	4.1E-18	1	1	no
JAK1	6.7E-03	4.6E-03	8.3E-10	3.0E-03	1.4E-07	1	2	no
JAK3	1.2E-01	4.2E-03	1.2E-01	1.4E-01	2.6E-02	1	1	no
NFAT	1.3E-03	8.1E-04	3.2E-03	4.3E-18	1.3E-18	2	1	no
RORgt	1.5E-01	3.3E-02	9.1E-02	1.3E-01	2.4E-02	6	7	yes
SOCS1	2.7E-02	1.2E-03	2.9E-02	9.1E-02	4.8E-03	2	2	no
STAT1	3.6E-02	5.5E-03	3.3E-02	2.6E-03	1.6E-09	3	2	no
STAT3	4.1E-02	1.7E-03	2.5E-02	1.1E-02	1.1E-03	2	2	no
STAT4	5.4E-06	7.9E-20	3.2E-06	5.3E-02	3.5E-18	1	2	no
STAT5	8.4E-03	1.9E-03	1.6E-03	4.6E-17	1.2E-17	1	1	no
STAT6	4.8E-03	2.8E-03	4.6E-03	3.5E-02	1.2E-03	3	1	no
Tbet	2.6E-02	1.4E-02	3.2E-03	9.1E-02	4.8E-03	6	5	yes
TCR	1.7E-17	8.3E-18	5.9E-18	1.3E-19	4.2E-20	1	0	no
TGFb	4.8E-03	3.6E-03	5.9E-17	1.3E-19	4.1E-20	1	0	no
TGFbR	4.8E-03	2.7E-03	3.2E-03	4.1E-18	1.2E-18	2	1	no

7.4.2 T-helper differentiation (with input loops)

7.4.2.1 Total centrality



7.4.2.2 Dynamic centrality



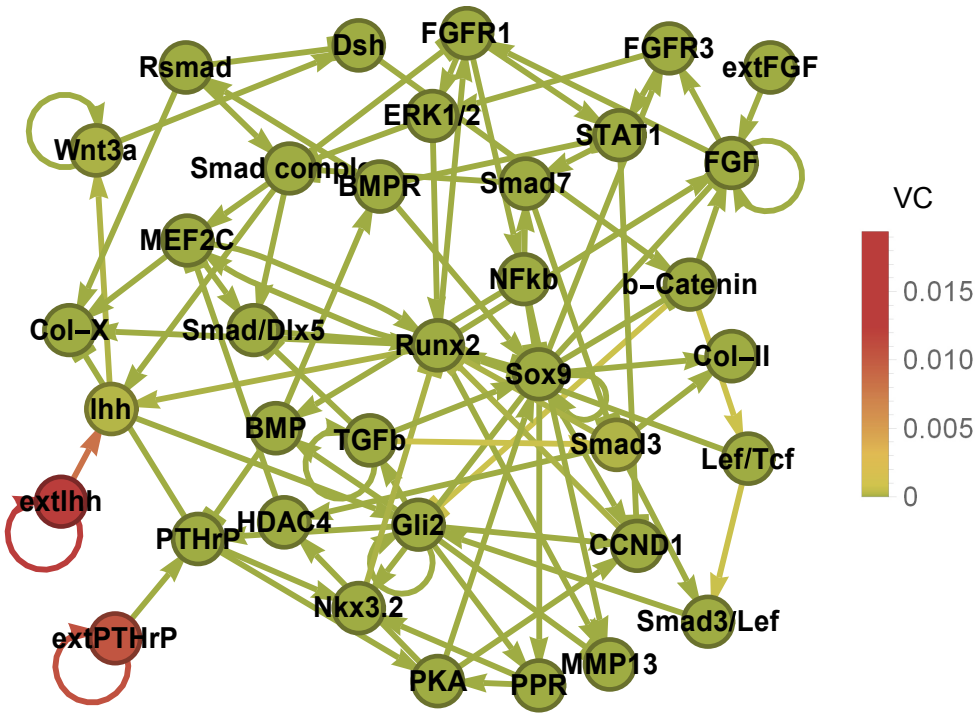
7.4.2.3 Additional data

Node	TC	VC	DC	TC-Sens.	VC-Sens.	Outgoing	Incoming	Loop
Foxp3	4.0E-02	2.3E-02	6.8E-04	7.0E-02	2.2E-02	4	6	yes
GATA3	2.0E-02	8.3E-03	1.4E-04	4.0E-03	2.1E-04	7	5	yes
IFNb	2.0E-01	7.9E-02	2.5E-03	2.2E-02	8.9E-03	2	1	yes
IFNbR	1.4E-01	1.1E-03	1.3E-01	2.2E-02	8.9E-03	1	1	no
IFNg	5.2E-03	1.7E-04	4.1E-03	6.1E-02	8.4E-03	1	5	no
IFNgR	4.2E-04	3.1E-04	2.0E-07	6.9E-02	9.0E-03	1	1	no
IL10	7.8E-03	2.6E-03	8.1E-03	3.9E-03	1.8E-04	1	1	no
IL10R	8.1E-03	1.4E-03	2.1E-03	4.0E-03	1.8E-04	1	1	no
IL12	3.3E-02	2.7E-02	2.3E-02	2.1E-02	9.5E-03	2	1	yes
IL12R	9.7E-03	5.7E-21	9.2E-03	5.8E-02	1.5E-02	1	2	no
IL17	0.0E+00	9.2E-23	0.0E+00	1.1E-01	3.1E-02	0	1	no
IL18	3.3E-02	2.6E-02	2.2E-02	2.2E-02	9.6E-03	2	1	yes
IL18R	1.8E-02	1.1E-18	2.0E-02	5.7E-02	1.5E-02	1	2	no
IL2	9.8E-02	4.4E-02	2.8E-03	2.1E-02	9.6E-03	2	1	yes
IL23	1.3E-01	5.3E-02	1.8E-16	2.0E-02	8.8E-03	2	1	yes
IL23R	8.7E-02	1.4E-03	8.1E-02	2.0E-02	8.8E-03	1	1	no
IL2R	5.7E-02	2.5E-03	5.1E-02	2.1E-02	9.6E-03	1	1	no
IL4	8.3E-03	1.6E-04	9.9E-03	5.2E-02	7.7E-03	1	3	no
IL4R	1.7E-02	1.4E-03	1.8E-02	6.0E-02	8.6E-03	1	2	no
IL6	7.3E-02	6.9E-03	5.7E-02	1.1E-01	3.1E-02	1	1	no
IL6R	7.0E-02	5.0E-03	7.5E-02	1.2E-01	3.2E-02	1	1	no
IRAK	3.0E-04	5.2E-18	6.3E-04	7.5E-02	1.5E-02	1	1	no
JAK1	4.0E-03	1.7E-03	1.1E-04	9.8E-03	1.0E-07	1	2	no
JAK3	3.3E-02	5.0E-03	2.3E-02	1.2E-01	3.2E-02	1	1	no
NFAT	2.6E-02	4.8E-05	1.8E-02	2.2E-02	9.9E-03	2	1	no
RORgt	1.0E-01	5.0E-02	1.8E-02	9.0E-02	2.5E-02	6	7	yes
SOCS1	1.2E-02	1.8E-16	1.4E-02	5.7E-02	7.9E-03	2	2	no
STAT1	1.5E-01	4.8E-03	1.4E-01	3.7E-02	7.7E-03	3	2	no
STAT3	7.3E-02	2.7E-03	6.1E-02	3.7E-02	7.7E-03	2	2	no
STAT4	2.1E-04	1.3E-19	6.3E-04	6.8E-02	1.5E-02	1	2	no
STAT5	2.6E-02	5.3E-03	2.8E-02	4.3E-02	9.6E-03	1	1	no
STAT6	1.5E-02	6.4E-04	8.6E-03	6.4E-02	8.7E-03	3	1	no
Tbet	2.7E-02	1.4E-02	2.8E-04	5.1E-02	1.5E-02	6	5	yes

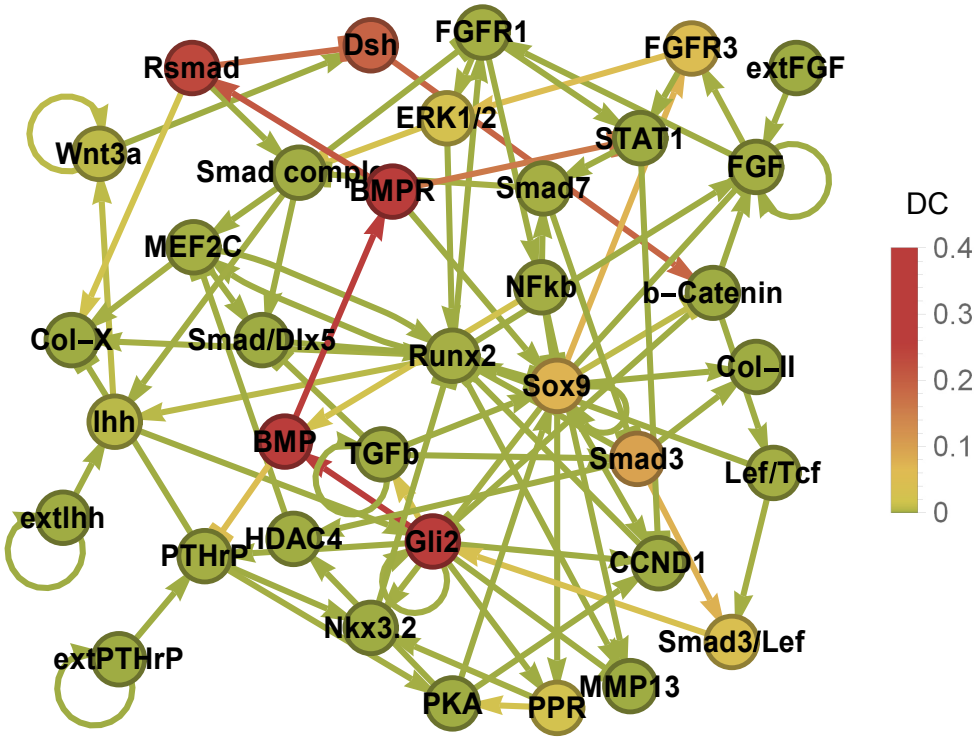
TCR	7.4E-02	3.1E-02	3.5E-19	2.2E-02	9.9E-03	2	1	yes
TGFb	7.2E-02	3.7E-02	1.6E-16	2.1E-02	1.1E-02	2	1	yes
TGFbR	2.1E-02	2.7E-03	1.2E-02	2.1E-02	1.1E-02	2	1	no

7.4.3 Chondrocyte proliferation and differentiation

7.4.3.1 Value centrality



7.4.3.2 Dynamic centrality



7.4.3.3 Additional data

Node	TC	VC	DC	TC-Sens.	VC-Sens.	Outgoing	Incoming	Loop
b-Catenin	2.8E-16	2.1E-16	1.6E-16	4.0E-02	1.3E-11	3	2	no
BMP	3.5E-01	1.1E-16	3.3E-01	3.5E-02	1.8E-07	2	2	no
BMPR	3.8E-01	1.6E-16	4.0E-01	6.3E-02	4.9E-10	3	1	no
CCND1	7.2E-11	1.1E-16	7.2E-11	1.1E-01	9.2E-11	1	4	no
Col-II	0.0E+00	9.8E-22	0.0E+00	4.7E-02	6.7E-10	0	2	no
Col-X	0.0E+00	6.2E-22	0.0E+00	6.6E-02	2.8E-08	0	4	no
Dsh	1.9E-01	1.0E-16	1.9E-01	1.1E-01	1.1E-14	1	2	no
ERK1/2	2.7E-02	1.1E-16	2.6E-02	4.6E-02	8.2E-11	2	2	no
extFGF	1.7E-16	1.3E-16	5.4E-17	5.8E-20	1.4E-20	1	0	no
extIhh	3.5E-02	1.9E-02	4.8E-07	2.0E-02	1.1E-02	2	1	yes
extPTHrP	2.1E-02	1.0E-02	8.8E-12	2.0E-02	1.0E-02	2	1	yes
FGF	1.5E-04	3.6E-16	1.5E-04	4.5E-02	3.2E-14	3	5	yes
FGFR1	2.5E-04	5.7E-16	2.5E-04	4.6E-02	3.7E-12	3	3	no
FGFR3	5.0E-02	9.7E-17	5.0E-02	2.4E-02	6.7E-10	2	2	no
Gli2	3.2E-01	1.7E-16	3.3E-01	6.4E-02	2.3E-06	7	4	yes
HDAC4	1.2E-05	1.2E-17	1.2E-05	4.1E-02	5.8E-08	1	2	no
Lef/Tcf	2.0E-03	8.2E-17	1.4E-04	4.0E-02	2.6E-13	2	1	no
Ihh	1.1E-02	1.4E-04	4.7E-03	3.7E-02	8.2E-03	2	3	no
MEF2C	2.1E-04	3.6E-17	2.1E-04	3.5E-02	3.4E-13	3	3	no
MMP13	0.0E+00	1.1E-21	0.0E+00	4.6E-02	3.8E-12	0	3	no
NFkb	1.4E-04	4.0E-16	1.4E-04	4.7E-02	8.2E-14	4	1	no
Nkx3.2	7.1E-11	1.0E-16	7.1E-11	4.0E-02	1.7E-08	1	3	no
PKA	2.6E-08	1.2E-16	2.6E-08	6.0E-02	5.2E-07	3	2	no
PPR	2.4E-02	9.5E-17	2.4E-02	4.7E-02	9.7E-10	2	2	no
PTHrP	1.4E-04	9.9E-17	1.4E-04	4.9E-02	4.8E-05	3	3	no
Rsmad	2.2E-01	1.2E-16	2.4E-01	8.8E-02	5.7E-13	3	1	no

Runx2	3.5E-04	4.3E-16	3.1E-04	4.0E-02	1.3E-10	7	7	no
Smad	1.4E-04	3.3E-16	4.6E-05	6.9E-02	1.7E-14	4	3	no
Smad/Dlx	3.6E-13	2.6E-17	3.6E-13	4.3E-03	1.0E-14	1	3	no
Smad3	9.5E-02	9.1E-05	9.7E-02	1.1E-01	3.6E-13	4	2	no
Smad3/Lef	2.9E-02	3.5E-17	3.9E-02	8.2E-02	2.8E-14	1	2	no
Smad7	2.9E-04	9.0E-17	2.8E-04	9.8E-02	1.2E-11	2	2	no
Sox9	7.5E-02	3.5E-16	7.4E-02	4.6E-02	5.8E-09	8	6	yes
STAT1	4.9E-12	1.8E-16	4.9E-12	7.6E-02	8.2E-11	2	3	no
TGFb	1.0E-05	1.2E-16	1.0E-05	1.2E-02	3.1E-10	4	2	yes
Wnt3a	5.4E-03	4.5E-05	6.1E-03	1.3E-02	5.8E-04	2	2	yes

7.4.4 *A. thaliana* inflorescence

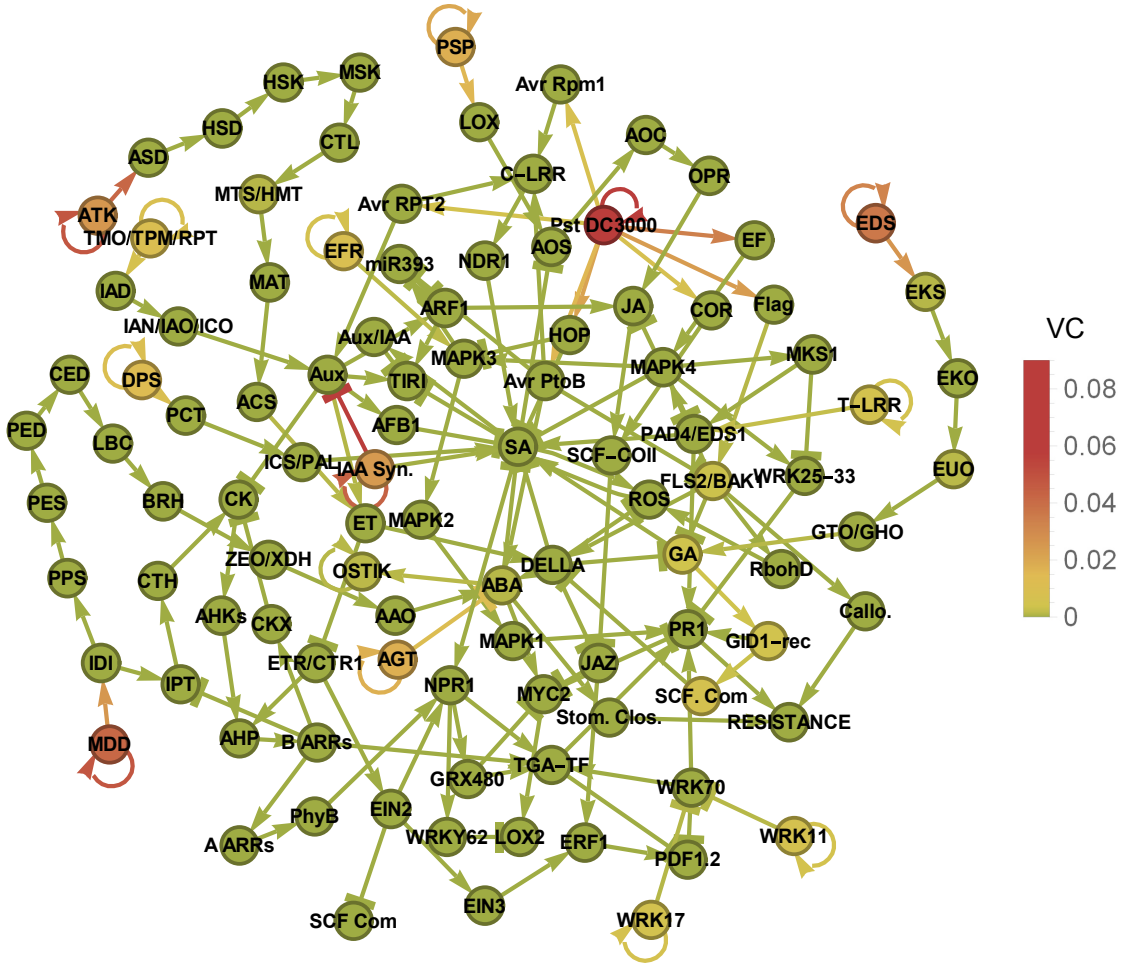
The complex Boolean functions cannot be displayed in a straightforward network graph.

7.4.4.1 Additional data

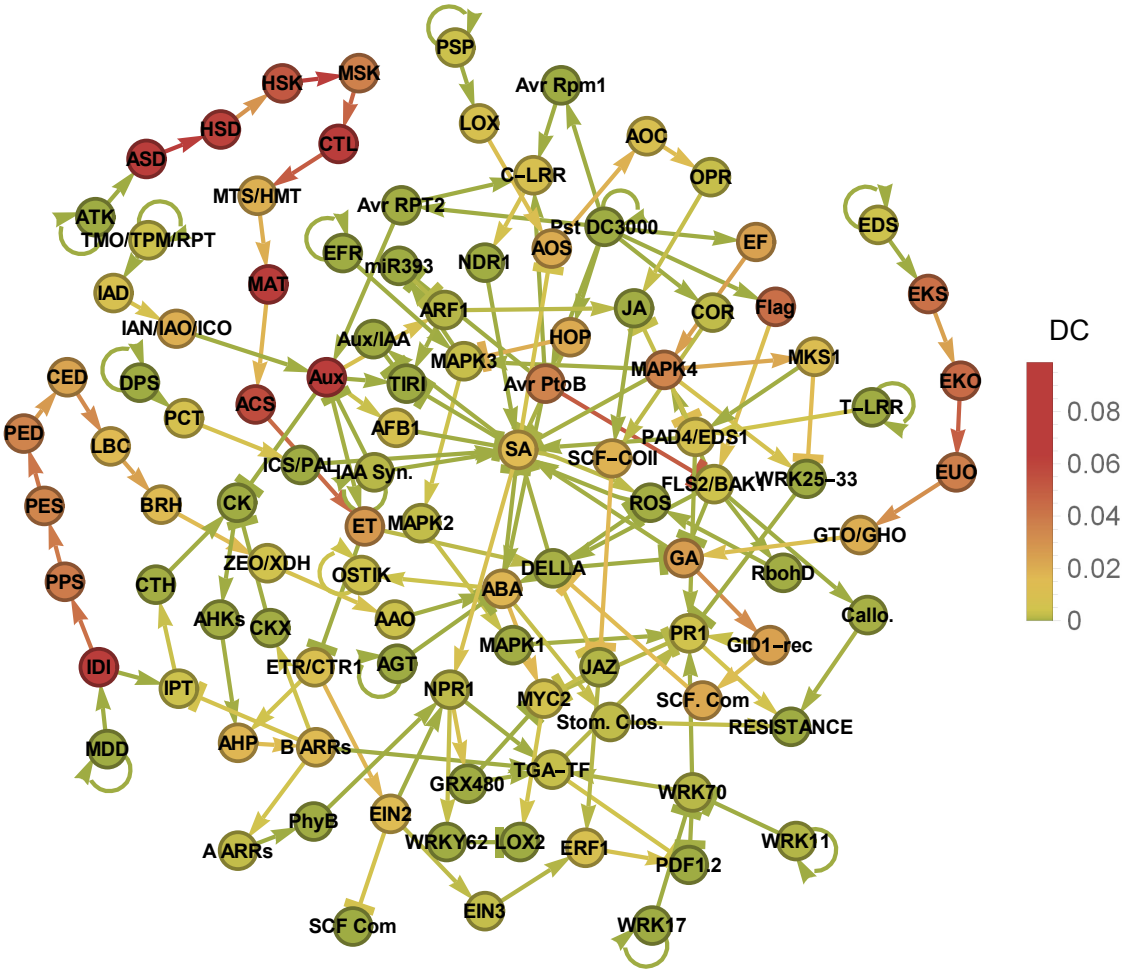
Node	**TC**	**VC**	**DC**	**TC-Sens.**	**VC-Sens.**	**Outgoing**	**Incoming**	**Loop**
AG	1.7E-01	4.0E-03	1.7E-01	5.5E-02	3.8E-03	5	8	yes
AP1	2.5E-02	2.3E-03	1.7E-02	8.4E-02	3.4E-03	5	6	no
AP2	1.5E-03	1.5E-03	1.8E-16	8.7E-02	4.6E-04	1	1	no
AP3	1.2E-02	3.5E-03	6.9E-03	8.4E-02	3.4E-02	3	7	yes
EMF1	1.6E-01	5.5E-03	2.0E-01	4.9E-02	4.6E-04	4	1	no
FT	1.5E-03	6.9E-17	2.3E-03	1.2E-01	4.6E-04	1	1	no
FUL	0.0E+00	3.0E-20	0.0E+00	9.4E-02	3.8E-03	0	2	no
LFY1	1.5E-01	6.3E-03	1.4E-01	4.9E-02	4.6E-04	7	2	no
PI	4.8E-03	3.4E-03	7.7E-04	6.6E-02	1.2E-03	3	6	yes
SEP	3.1E-03	1.7E-03	1.5E-03	1.3E-01	4.6E-04	4	1	no
TFL1	4.0E-01	1.5E-03	4.6E-01	4.9E-02	4.6E-04	5	3	no
UFO	7.0E-02	4.7E-02	3.8E-03	5.8E-02	2.8E-02	2	1	yes
WUS	1.4E-03	6.2E-04	2.1E-16	8.1E-02	4.6E-04	2	3	yes

7.4.5 *A. thaliana* immune response

7.4.5.1 Value centrality



7.4.5.2 Dynamic centrality



7.4.5.3 Additional data

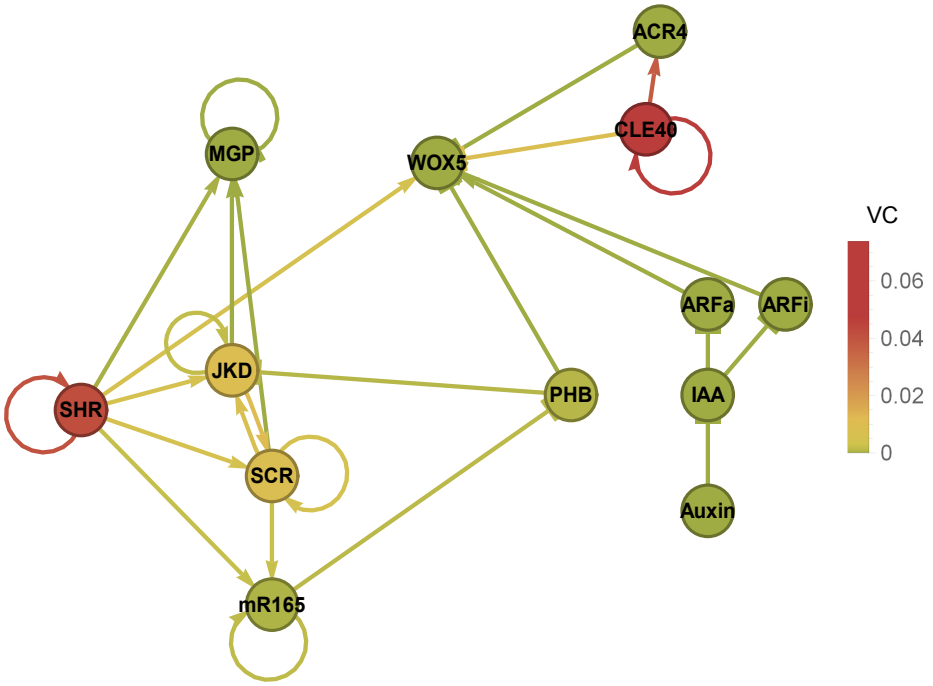
Node	TC	VC	DC	TC-Sens.	VC-Sens.	Outgoing	Incoming	Loop
A ARRs	2.8E-03	1.7E-20	2.1E-03	3.2E-02	1.0E-03	1	1	no
AAO	1.7E-04	1.8E-18	3.4E-03	5.0E-02	3.4E-03	1	1	no
ABA	1.5E-02	8.7E-04	1.8E-02	9.9E-03	3.7E-03	4	4	no
ACS	3.4E-02	2.4E-19	5.7E-02	4.1E-02	2.0E-03	1	1	no
AFB1	2.9E-03	1.5E-21	7.5E-03	9.5E-03	2.3E-03	1	1	no
AGT	3.2E-02	1.8E-02	1.3E-21	7.7E-03	3.3E-03	2	1	yes
AHKs	1.9E-05	2.5E-17	2.8E-05	5.0E-03	7.4E-07	1	1	no
AHP	2.6E-02	6.8E-18	1.7E-02	2.3E-02	9.1E-04	1	2	no
AOC	9.5E-03	5.0E-23	9.2E-03	2.8E-02	5.6E-03	1	1	no
AOS	1.8E-02	1.7E-21	2.1E-02	1.9E-02	5.0E-03	1	2	no
ARF1	8.5E-04	1.2E-21	9.0E-04	1.5E-02	3.9E-03	2	2	no
ASD	7.5E-02	5.2E-17	6.8E-02	5.0E-03	2.0E-03	1	1	no
ATK	7.9E-02	2.5E-02	1.1E-16	5.0E-03	2.0E-03	2	1	yes
Aux	3.1E-02	4.1E-17	7.0E-02	6.9E-03	2.3E-03	5	3	no
Aux/IAA	1.8E-06	2.5E-25	2.0E-06	1.5E-02	2.0E-03	1	1	no
Avr PtoB	4.5E-02	8.9E-18	3.6E-02	7.9E-03	2.7E-03	4	1	no
Avr Rpm1	2.6E-21	7.9E-21	1.3E-27	7.9E-03	2.7E-03	1	1	no
Avr RPT2	7.7E-03	2.7E-20	1.6E-05	7.9E-03	2.7E-03	2	1	no

B ARRs	1.5E-02	1.2E-17	1.5E-02	3.0E-02	1.0E-03	4	1	no
BRH	9.9E-03	1.4E-21	1.4E-02	4.0E-02	3.4E-03	1	1	no
Callo.	8.6E-05	1.1E-24	1.1E-04	9.6E-03	5.5E-23	1	1	no
CED	2.7E-02	1.3E-21	2.3E-02	2.8E-02	3.4E-03	1	1	no
CK	1.2E-06	5.0E-17	1.2E-06	5.1E-03	9.0E-06	1	3	no
CKX	1.9E-05	2.7E-17	1.7E-05	3.2E-02	1.0E-03	1	1	no
C-LRR	7.5E-03	2.5E-23	8.4E-03	7.9E-03	2.7E-03	1	3	no
COR	2.6E-04	8.4E-21	1.8E-03	7.9E-03	2.7E-03	1	1	no
CTH	4.1E-05	1.3E-17	3.1E-05	1.4E-02	1.7E-03	1	1	no
CTL	5.4E-02	5.0E-18	7.8E-02	2.5E-02	2.0E-03	1	1	no
DELLA	4.2E-03	9.2E-18	1.3E-04	4.2E-02	5.1E-03	5	3	no
DPS	2.2E-02	1.3E-02	7.9E-20	7.2E-03	4.2E-03	2	1	yes
EDS	6.9E-02	3.6E-02	3.6E-03	7.3E-03	3.7E-03	2	1	yes
EF	2.4E-02	2.8E-20	2.4E-02	7.9E-03	2.7E-03	1	1	no
EFR	1.6E-02	8.8E-03	1.1E-22	7.3E-03	4.7E-03	2	1	yes
EIN2	1.2E-02	2.3E-21	1.4E-02	2.6E-02	1.0E-03	3	1	no
EIN3	2.2E-03	3.7E-23	2.3E-03	3.0E-02	1.1E-03	1	1	no
EKO	4.4E-02	4.5E-19	4.4E-02	1.4E-02	3.7E-03	1	1	no
EKS	5.3E-02	7.9E-04	4.2E-02	7.3E-03	3.7E-03	1	1	no
ERF1	4.6E-03	3.9E-26	8.1E-03	3.9E-02	3.7E-03	1	2	no
ET	2.2E-02	5.5E-17	2.7E-02	2.1E-02	9.2E-04	2	2	no
ETR/CTR	3.4E-03	1.4E-17	9.5E-03	2.6E-02	1.0E-03	2	1	no
EUO	4.0E-02	1.1E-03	3.7E-02	2.2E-02	3.7E-03	1	1	no
Flag	4.7E-02	3.0E-17	4.3E-02	7.9E-03	2.7E-03	1	1	no
FLS2/BA	3.0E-03	3.1E-03	3.8E-03	9.6E-03	2.6E-23	5	2	no
GA	3.6E-02	3.7E-03	2.6E-02	5.0E-02	6.2E-03	2	3	no
GID1-rec	3.0E-02	4.1E-03	2.4E-02	5.5E-02	6.6E-03	2	1	no
GRX480	1.3E-03	6.0E-23	2.4E-07	3.3E-02	2.1E-03	2	1	no
GTO/GH	1.2E-02	8.3E-18	2.0E-02	2.9E-02	3.7E-03	1	1	no
HOP	2.4E-02	2.7E-21	2.1E-02	7.9E-03	2.7E-03	2	1	no
HSD	4.6E-02	5.1E-17	6.1E-02	9.9E-03	2.0E-03	1	1	no
HSK	6.0E-02	1.2E-17	5.2E-02	1.5E-02	2.0E-03	1	1	no
IAA Syn.	7.0E-02	2.4E-02	1.0E-03	5.4E-03	2.6E-03	3	1	yes
IAD	1.9E-02	3.3E-18	8.4E-03	6.6E-03	3.1E-03	1	1	no
IAN/IAO/I	2.7E-03	8.5E-19	2.0E-02	1.4E-02	3.1E-03	1	1	no
ICS/PAL	2.3E-05	8.0E-22	4.2E-08	1.5E-02	4.2E-03	1	1	no
IDI	6.7E-02	2.9E-17	7.7E-02	4.7E-03	3.4E-03	2	1	no
IPT	3.1E-03	5.2E-19	3.5E-03	1.1E-02	1.7E-03	1	2	no
JA	1.8E-03	5.8E-22	8.8E-06	3.4E-02	4.7E-03	1	3	no
JAZ	2.1E-03	3.5E-24	6.5E-04	2.3E-02	3.3E-03	2	2	no
LBC	1.5E-02	2.2E-22	1.6E-02	3.5E-02	3.4E-03	1	1	no
LOX	6.0E-03	4.2E-22	7.5E-03	7.3E-03	3.6E-03	1	1	no
LOX2	0.0E+00	3.6E-26	0.0E+00	2.1E-02	3.3E-03	0	2	no
MAPK1	2.5E-05	3.8E-23	1.6E-05	2.2E-02	2.9E-03	1	1	no
MAPK2	2.8E-03	7.3E-23	2.1E-03	1.9E-02	2.9E-03	1	1	no
MAPK3	5.3E-03	1.6E-21	2.7E-03	1.5E-02	2.5E-03	2	3	no
MAPK4	3.1E-02	5.6E-23	3.5E-02	1.6E-02	2.4E-03	5	2	no
MAT	5.8E-02	1.0E-17	8.0E-02	3.5E-02	2.0E-03	1	1	no
MDD	5.3E-02	4.1E-02	5.0E-18	4.7E-03	3.4E-03	2	1	yes
miR393	5.4E-04	9.3E-23	3.1E-22	9.8E-03	2.9E-03	1	2	no
MKS1	1.1E-02	2.3E-23	1.1E-02	2.3E-02	2.7E-03	2	1	no
MSK	5.2E-02	1.2E-17	3.7E-02	2.1E-02	2.0E-03	1	1	no
MTS/HM	3.4E-02	8.5E-04	1.9E-02	3.1E-02	2.0E-03	1	1	no
MYC2	4.7E-03	2.8E-22	5.6E-03	1.6E-02	3.1E-03	2	3	no
NDR1	5.5E-06	3.0E-23	4.1E-06	1.5E-02	2.7E-03	1	1	no
NPR1	7.6E-03	2.9E-19	1.6E-03	2.7E-02	1.8E-03	3	3	no
OPR	2.8E-03	3.7E-23	2.6E-03	3.4E-02	5.8E-03	1	1	no
OSTIK	5.8E-03	1.2E-03	3.8E-03	7.8E-03	2.4E-03	1	2	yes
PAD4/ED	2.1E-03	5.7E-22	7.0E-03	1.7E-02	2.5E-03	3	3	no

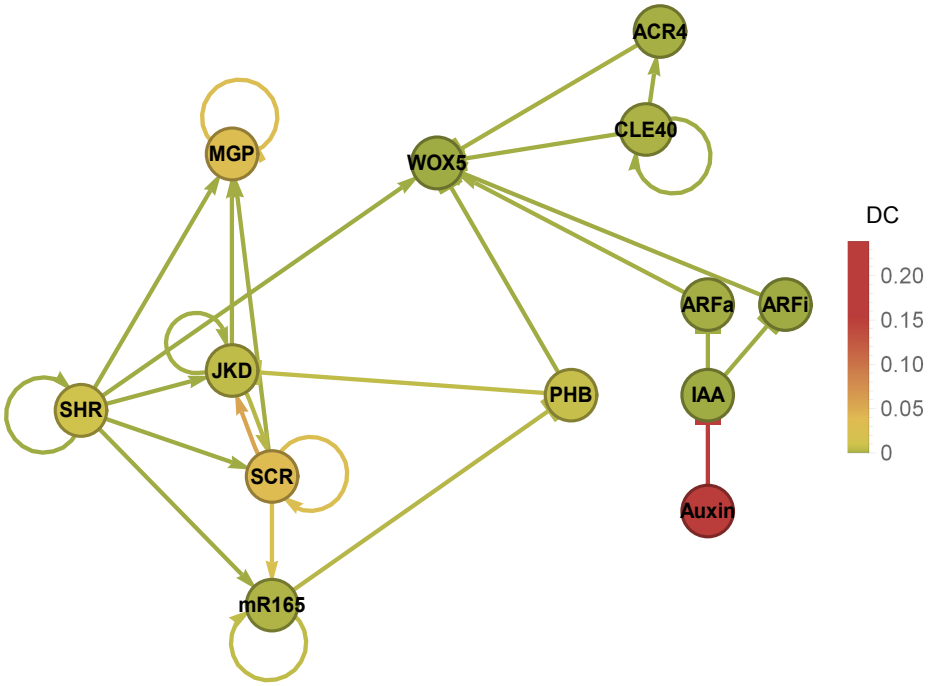
PCT	7.7E-03	1.9E-22	6.7E-03	7.2E-03	4.2E-03	1	1	no
PDF1.2	0.0E+00	1.7E-25	0.0E+00	2.0E-02	2.4E-03	0	3	no
PED	3.0E-02	1.0E-19	3.6E-02	2.3E-02	3.4E-03	1	1	no
PES	3.7E-02	1.9E-18	3.5E-02	1.7E-02	3.4E-03	1	1	no
PhyB	1.1E-05	8.0E-21	1.7E-06	3.5E-02	1.1E-03	1	1	no
PPS	3.4E-02	3.8E-17	3.8E-02	1.2E-02	3.4E-03	1	1	no
PR1	3.7E-03	1.6E-23	4.2E-03	3.8E-02	5.1E-03	1	7	no
PSP	3.7E-02	1.8E-02	3.2E-03	7.3E-03	3.6E-03	2	1	yes
Pst	1.8E-01	6.2E-02	1.2E-16	7.9E-03	2.7E-03	8	1	yes
RbohD	6.1E-06	2.0E-21	1.2E-05	9.6E-03	5.4E-23	1	1	no
RESISTA	0.0E+00	2.7E-26	0.0E+00	2.8E-02	2.8E-03	0	3	no
ROS	2.9E-06	5.6E-22	3.6E-05	1.5E-02	2.0E-03	1	3	no
SA	8.2E-03	3.1E-19	1.5E-02	1.3E-02	2.0E-03	5	11	no
SCF Com	0.0E+00	6.2E-27	0.0E+00	3.2E-02	1.1E-03	0	1	no
SCF. Com	3.2E-02	6.0E-03	2.2E-02	5.8E-02	6.8E-03	1	1	no
SCF-COII	1.3E-02	5.7E-24	1.8E-02	1.3E-02	3.2E-03	1	2	no
Stom.	5.3E-04	2.6E-24	1.9E-03	1.4E-02	4.5E-03	1	1	no
TGA-TF	3.8E-03	2.7E-24	2.6E-03	3.0E-02	3.2E-03	2	4	no
TIRI	0.0E+00	8.3E-27	0.0E+00	1.4E-02	2.7E-03	0	4	no
T-LRR	1.3E-02	5.1E-03	6.8E-09	6.7E-03	3.4E-03	2	1	yes
TMO/TP	2.8E-02	9.2E-03	3.5E-03	6.6E-03	3.1E-03	2	1	yes
WRK11	1.0E-02	5.6E-03	6.8E-04	7.0E-03	3.8E-03	2	1	yes
WRK17	1.0E-02	5.4E-03	2.5E-20	6.5E-03	3.4E-03	2	1	yes
WRK25-	7.5E-07	3.4E-23	1.4E-06	1.5E-02	2.6E-11	1	2	no
WRK70	1.0E-03	2.1E-23	1.0E-03	3.4E-03	1.8E-03	3	2	no
WRKY62	5.7E-08	1.9E-25	8.6E-08	3.3E-02	2.1E-03	1	1	no
ZEO/XDH	6.1E-03	2.3E-18	4.8E-03	4.5E-02	3.4E-03	1	1	no

7.4.6 *A. thaliana* root stem cell niche

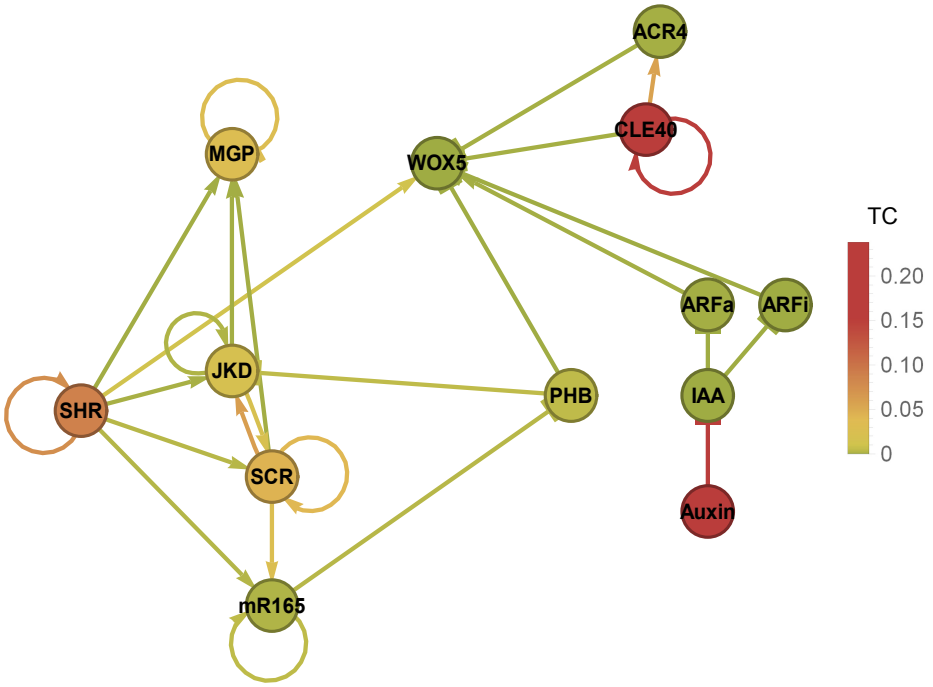
7.4.6.1 Value centrality



7.4.6.2 Dynamic centrality



7.4.6.3 Total centrality



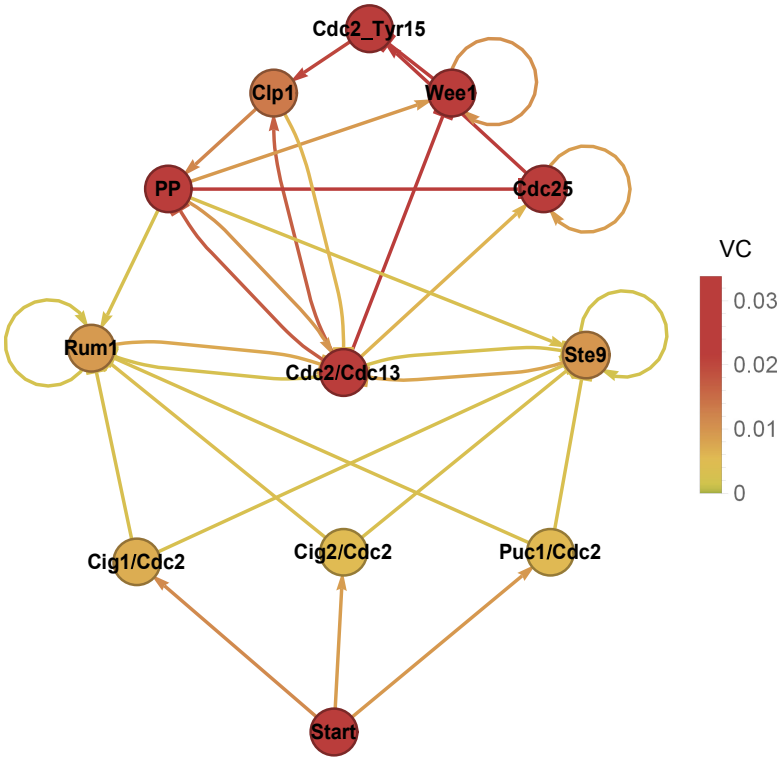
7.4.6.4 Additional data

Node	TC	VC	DC	TC-Sens.	VC-Sens.	Outgoing	Incoming	Loop
ACR4	1.4E-04	1.8E-17	1.2E-04	6.0E-02	2.8E-02	1	1	no

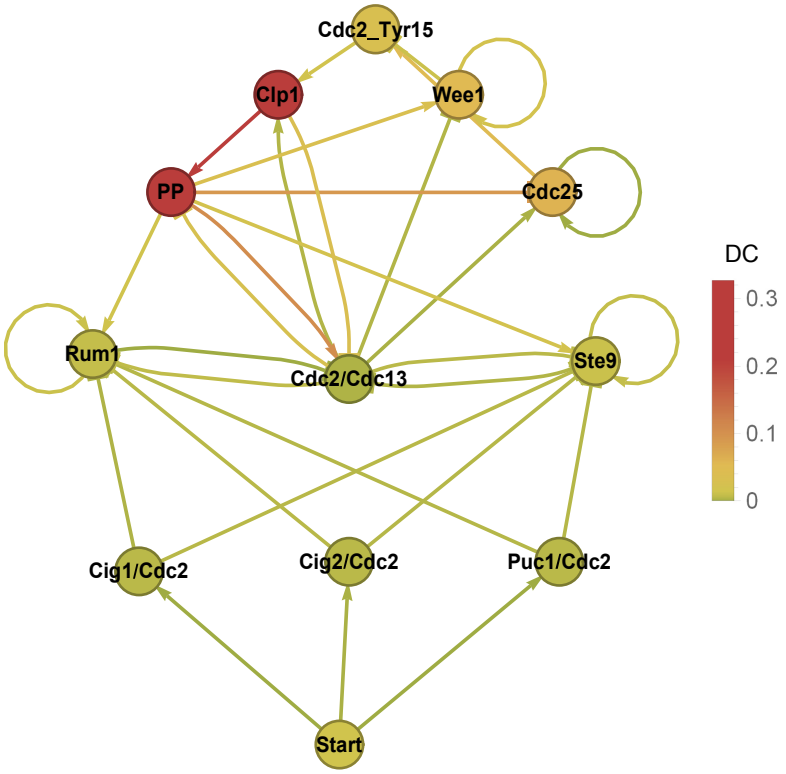
ARFa	6.9E-05	1.6E-17	8.0E-05	7.7E-02	1.9E-16	1	1	no
ARFi	2.7E-06	3.6E-17	3.0E-06	7.7E-02	1.7E-16	1	1	no
Auxin	2.4E-01	2.3E-16	2.4E-01	1.8E-17	7.6E-18	1	0	no
CLE40	1.6E-01	7.4E-02	7.1E-04	6.0E-02	2.8E-02	3	1	yes
IAA	1.7E-16	8.3E-17	1.3E-16	7.7E-02	1.8E-16	2	1	no
JKD	1.9E-02	9.0E-03	4.4E-03	2.6E-02	6.6E-03	3	4	yes
MGP	3.0E-02	5.8E-21	3.0E-02	3.3E-02	8.7E-04	1	4	yes
mR165	1.1E-03	3.1E-04	1.1E-03	1.5E-02	6.6E-03	2	3	yes
PHB	4.0E-03	6.2E-04	6.2E-03	1.6E-02	6.8E-03	2	1	no
SCR	4.3E-02	8.4E-03	3.2E-02	2.6E-02	6.6E-03	4	3	yes
SHR	8.9E-02	4.2E-02	1.0E-02	5.8E-02	2.7E-02	6	1	yes
WOX5	0.0E+00	3.3E-18	0.0E+00	5.7E-02	2.4E-02	0	6	no

7.4.7 *S. pompe* (fission yeast) cell cycle

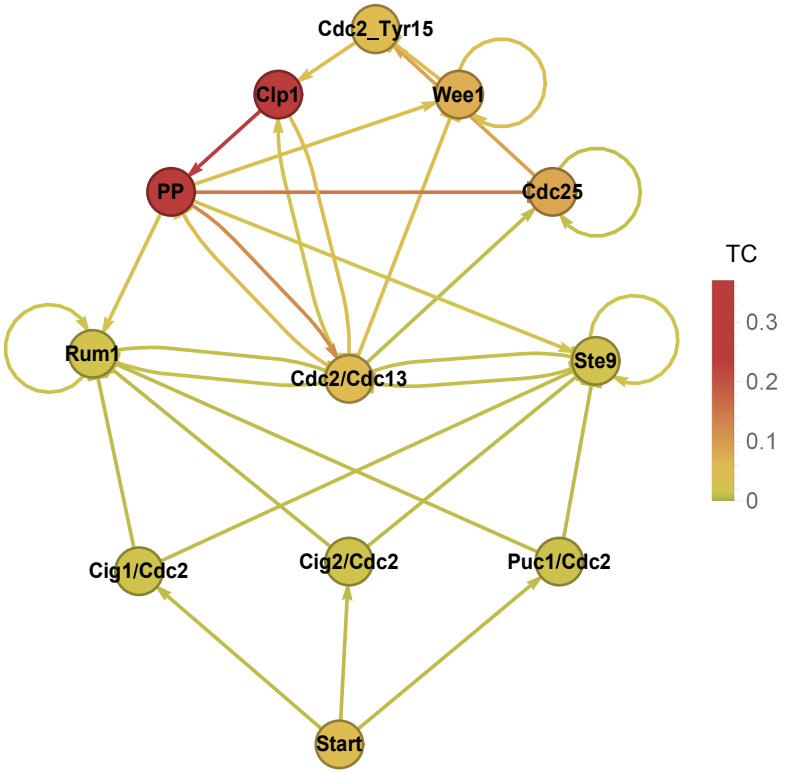
7.4.7.1 Value centrality



7.4.7.2 Dynamic centrality



7.4.7.3 Total centrality



7.4.7.4 Additional data

Node	TC	VC	DC	TC-Sens.	VC-Sens.	Outgoing	Incoming	Loop
Cdc2/Cdc	5.5E-02	3.4E-02	1.3E-03	2.5E-03	7.2E-14	6	4	no
Cdc2_Tyr	5.0E-02	2.2E-02	3.0E-02	8.5E-02	2.0E-08	1	2	no
Cdc25	8.4E-02	2.4E-02	6.0E-02	8.0E-02	7.3E-04	2	3	yes
Cig1/Cdc2	1.4E-02	6.9E-03	4.0E-03	2.2E-17	5.1E-18	2	1	no
Cig2/Cdc2	1.4E-02	5.2E-03	4.0E-03	2.1E-17	5.2E-18	2	1	no
Clp1	2.5E-01	1.3E-02	2.2E-01	7.7E-02	2.2E-10	2	2	no
PP	3.7E-01	2.7E-02	3.3E-01	1.5E-01	3.9E-12	5	2	no
Puc1/Cdc2	1.2E-02	5.6E-03	4.0E-03	2.1E-17	5.1E-18	2	1	no
Rum1	1.9E-02	9.2E-03	7.3E-03	1.9E-01	5.7E-02	2	6	yes
Start	4.2E-02	2.7E-02	1.3E-02	7.4E-19	1.8E-19	3	0	no
Ste9	1.9E-02	9.7E-03	1.0E-02	2.0E-01	5.8E-02	2	6	yes
Wee1	7.2E-02	2.4E-02	5.3E-02	2.2E-01	9.3E-02	2	3	yes

7.4.8 *S. cerevisiae* (budding yeast)

Node	TC	VC	DC	TC-Sens.	VC-Sens.	Outgoing	Incoming	Loop
abf1	2.1E-02	1.6E-02	1.9E-02	7.6E-03	3.5E-03	9	1	yes
ace2	1.5E-03	3.3E-20	7.3E-04	1.7E-02	2.1E-18	2	2	no
adr1	1.2E-03	8.9E-18	2.4E-03	2.4E-05	7.1E-06	2	5	no
aft1	1.1E-03	8.7E-20	9.4E-04	1.0E-02	3.9E-21	2	2	yes
aft2	9.5E-03	3.9E-21	1.0E-02	2.0E-02	5.7E-04	1	1	no
arg81	1.4E-03	4.4E-34	1.4E-03	1.4E-03	4.4E-34	1	1	yes
aro80	2.2E-20	4.3E-20	7.9E-19	3.1E-18	7.4E-19	2	2	yes
ash1	3.1E-05	2.4E-21	8.7E-08	5.5E-03	1.8E-03	2	4	no
cad1	1.1E-02	7.4E-03	1.0E-02	6.7E-03	3.3E-03	3	1	yes
cat8	0.0E+00	4.1E-23	0.0E+00	8.0E-03	9.8E-04	0	2	no
cbf1	1.3E-02	4.5E-04	9.2E-03	8.8E-03	1.9E-03	8	2	yes
cin5	3.6E-05	1.3E-20	1.7E-04	2.1E-03	2.0E-05	5	9	yes
dal80	4.7E-07	7.7E-23	3.7E-07	8.7E-05	1.3E-12	2	5	yes
dal81	0.0E+00	2.3E-23	0.0E+00	2.2E-03	3.5E-03	0	1	no
dal82	8.6E-03	4.6E-03	3.0E-04	6.1E-03	3.3E-03	2	1	yes
dig1	0.0E+00	2.8E-22	0.0E+00	3.9E-03	8.1E-20	0	3	no
fhl1	9.9E-03	5.7E-03	1.7E-05	7.6E-03	3.1E-03	6	1	yes
fkh1	1.4E-02	5.4E-03	3.3E-06	8.0E-03	3.1E-03	3	1	yes
fkh2	1.0E-02	1.9E-19	1.0E-02	9.8E-03	7.8E-19	2	1	no
flo8	1.9E-07	8.8E-22	2.9E-07	1.5E-18	7.2E-19	2	2	no
gal4	1.4E-04	3.9E-22	1.4E-04	3.9E-18	1.1E-18	1	2	no
gat1	6.2E-07	6.8E-20	1.3E-06	5.6E-04	7.5E-06	2	11	yes
gcn4	3.5E-04	7.9E-04	4.0E-05	7.8E-03	9.4E-09	12	3	no
gcr2	7.8E-03	2.9E-03	4.5E-08	7.8E-03	2.9E-03	2	1	yes
gln3	3.8E-09	2.4E-17	1.2E-08	9.1E-06	1.2E-10	5	4	yes
gzf3	8.4E-09	1.7E-22	9.9E-09	1.9E-03	8.1E-07	1	1	no
hal9	7.1E-03	5.7E-03	1.1E-03	7.1E-03	4.5E-03	2	1	yes
hap1	1.4E-04	1.0E-23	5.4E-06	6.8E-04	1.8E-04	3	7	yes
hap2	7.5E-03	2.4E-03	9.9E-04	7.5E-03	2.4E-03	2	1	yes
hap4	1.6E-02	1.6E-04	1.6E-02	1.4E-05	4.7E-07	1	10	no
hsf1	1.1E-02	5.0E-03	1.4E-03	7.6E-03	4.3E-03	3	1	yes
ifh1	8.0E-03	3.9E-03	9.8E-04	8.0E-03	3.9E-03	1	1	yes
ime1	7.0E-10	3.1E-23	3.0E-11	7.7E-08	1.6E-08	1	5	yes
ino2	0.0E+00	4.9E-22	0.0E+00	5.1E-18	1.2E-18	0	1	no
ino4	4.9E-05	2.8E-19	4.9E-05	4.1E-18	9.6E-19	4	2	yes
ixr1	1.7E-07	3.9E-23	1.5E-07	2.8E-04	6.5E-20	2	8	yes
leu3	4.4E-05	1.2E-18	3.7E-05	1.5E-07	9.8E-11	1	3	no

mbp1	1.0E-02	4.9E-03	1.1E-03	7.6E-03	3.7E-03	3	1	yes
mcm1	9.2E-03	3.9E-03	7.2E-05	8.4E-03	3.5E-03	6	1	yes
met28	3.0E-08	9.0E-22	2.9E-08	3.3E-03	5.7E-04	1	12	yes
met31	7.3E-03	3.3E-03	2.0E-03	7.3E-03	3.3E-03	2	1	yes
met32	1.5E-05	2.7E-22	4.9E-05	2.4E-03	2.9E-04	1	3	no
met4	1.4E-03	1.6E-04	8.1E-04	8.7E-03	3.8E-03	5	4	yes
mga1	5.8E-06	3.5E-21	3.0E-06	1.7E-03	8.5E-04	1	10	yes
mig1	7.2E-20	2.0E-19	2.2E-19	4.0E-18	1.2E-18	4	2	yes
mot3	0.0E+00	3.7E-23	0.0E+00	1.6E-03	1.6E-04	0	4	no
msn1	0.0E+00	3.0E-23	0.0E+00	2.0E-02	5.7E-04	0	1	no
msn2	1.0E-02	1.4E-16	9.3E-03	2.9E-03	7.3E-04	28	6	yes
msn4	2.6E-03	6.8E-18	3.3E-04	6.6E-04	1.8E-04	5	10	yes
ndt80	0.0E+00	4.9E-22	0.0E+00	6.2E-19	2.3E-19	0	1	no
nrg1	6.4E-06	9.0E-23	5.3E-06	2.7E-03	6.0E-04	1	9	no
oaf1	1.5E-02	7.9E-03	1.0E-02	7.5E-03	2.7E-03	4	1	yes
oaf3	7.5E-03	4.1E-03	6.5E-19	7.5E-03	4.1E-03	1	1	yes
pdr1	7.7E-06	1.8E-17	9.2E-06	5.5E-03	1.1E-05	5	3	no
pdr3	3.8E-03	1.4E-19	3.4E-03	4.0E-03	3.9E-05	4	5	yes
phd1	1.9E-06	1.6E-23	3.5E-06	1.7E-04	3.3E-05	2	14	yes
pho2	8.0E-03	4.1E-03	9.8E-04	8.0E-03	4.1E-03	2	1	yes
pho4	3.1E-03	6.6E-04	1.3E-03	8.3E-03	3.1E-03	2	4	no
pip2	1.9E-05	5.7E-17	1.9E-05	3.8E-03	2.7E-03	1	3	yes
put3	0.0E+00	2.0E-22	0.0E+00	6.0E-05	3.8E-06	0	4	no
rap1	2.1E-02	3.2E-04	2.0E-02	3.2E-18	6.5E-19	17	2	yes
rds1	2.5E-03	0.0E+00	2.6E-03	5.0E-03	1.2E-04	1	3	yes
rds2	1.2E-02	5.4E-03	1.0E-03	7.1E-03	3.3E-03	3	1	yes
reb1	4.7E-04	1.6E-24	1.1E-03	1.3E-03	3.7E-04	1	4	yes
rfx1	1.7E-07	1.6E-20	3.1E-08	2.9E-18	1.1E-18	3	2	yes
rgm1	0.0E+00	7.1E-23	0.0E+00	6.6E-19	2.3E-19	0	1	no
rgt1	0.0E+00	3.5E-22	0.0E+00	6.4E-19	2.2E-19	0	1	no
rlm1	0.0E+00	4.1E-22	0.0E+00	3.1E-03	3.3E-03	0	1	no
rme1	0.0E+00	6.4E-23	0.0E+00	7.4E-03	1.3E-04	0	3	no
rox1	2.3E-07	6.2E-21	5.9E-08	6.9E-05	5.7E-06	5	13	yes
rpn4	3.4E-02	1.5E-18	3.2E-02	2.9E-03	7.9E-04	6	5	no
rtg3	0.0E+00	3.0E-23	0.0E+00	1.2E-02	2.4E-03	0	4	no
sfp1	1.2E-03	3.1E-04	8.1E-04	4.1E-18	1.2E-18	4	2	no
sip4	0.0E+00	3.0E-22	0.0E+00	7.4E-03	2.6E-03	0	3	no
skn7	2.6E-05	1.3E-18	2.3E-04	2.9E-03	1.4E-03	8	2	no
sko1	1.5E-02	9.2E-03	4.3E-03	6.7E-03	4.3E-03	8	1	yes
smp1	8.8E-04	6.4E-25	2.8E-04	1.3E-03	2.1E-04	1	5	yes
sok2	1.4E-02	9.8E-04	1.3E-02	3.5E-03	9.8E-04	26	5	yes
spt23	1.4E-03	2.8E-28	1.4E-03	1.4E-03	4.7E-34	2	1	yes
stb5	5.7E-03	8.2E-24	5.8E-03	5.0E-03	7.2E-05	2	2	yes
ste12	5.0E-03	1.5E-16	6.4E-03	6.0E-19	2.6E-19	36	4	yes
stp1	3.2E-04	1.1E-18	6.4E-10	2.2E-03	3.5E-03	1	1	no
sum1	1.2E-13	3.7E-21	6.3E-14	5.2E-04	4.8E-20	2	3	yes
sut1	5.0E-04	1.9E-21	6.5E-04	1.1E-03	1.1E-04	2	5	yes
swi4	9.3E-03	1.1E-19	9.3E-03	1.6E-03	3.6E-04	11	7	yes
swi5	7.7E-05	3.8E-17	1.2E-04	6.3E-03	2.2E-03	3	8	no
swi6	3.1E-05	3.2E-19	5.6E-05	6.5E-19	2.3E-19	3	1	no
tec1	1.6E-03	1.1E-16	4.3E-04	6.3E-19	3.8E-19	16	6	yes
thi2	8.4E-03	3.7E-03	8.9E-22	8.4E-03	3.7E-03	1	1	yes
tos8	0.0E+00	2.5E-23	0.0E+00	2.9E-03	1.1E-03	0	11	no
tup1	1.7E-02	9.6E-03	5.4E-03	6.9E-03	3.9E-03	6	1	yes
tye7	2.5E-05	4.2E-04	2.6E-05	1.6E-03	9.7E-05	1	15	no
ume6	8.3E-04	2.6E-21	8.6E-06	3.6E-03	1.6E-03	2	3	no
xbp1	5.9E-06	8.6E-23	2.5E-08	1.9E-03	3.8E-04	1	9	no
yap1	3.2E-02	9.3E-17	2.8E-02	1.0E-02	4.4E-04	15	4	yes
yap6	1.2E-03	2.4E-21	1.2E-03	2.0E-03	4.4E-04	2	5	no

yap7	1.4E-03	3.6E-34	1.4E-03	1.4E-03	3.6E-34	1	1	yes
yhp1	0.0E+00	2.0E-22	0.0E+00	1.2E-03	2.4E-04	0	6	no
yox1	2.7E-06	1.7E-20	3.0E-06	8.5E-03	4.3E-05	2	3	no
yrm1	1.2E-03	7.8E-21	1.2E-03	1.0E-02	2.2E-03	1	2	yes
yrr1	2.8E-08	1.2E-22	8.2E-09	3.0E-03	6.1E-06	1	2	yes
zap1	3.9E-20	2.4E-20	4.6E-21	3.8E-18	1.5E-18	2	2	yes

7.4.9 P. aeruginosa

Node	**TC**	**VC**	**DC**	**TC-Sens.**	**VC-Sens.**	**Outgoing**	**Incoming**	**Loop**
agmr	0.0E+00	3.2E-21	0.0E+00	2.2E-02	4.1E-03	0	1	no
agur	8.3E-03	4.8E-03	4.6E-03	8.3E-03	4.8E-03	1	1	yes
algq	3.9E-02	2.1E-02	2.4E-03	9.0E-03	4.1E-03	4	1	yes
algr	1.9E-04	1.9E-04	6.4E-07	1.2E-02	1.9E-03	2	3	yes
algr3	9.2E-03	3.5E-03	6.1E-04	8.3E-03	3.2E-03	2	1	yes
algr4	1.0E-02	6.5E-03	8.9E-06	8.3E-03	4.1E-03	2	1	yes
algu	3.5E-03	7.0E-17	4.3E-03	5.5E-04	7.2E-05	6	6	yes
algw	9.4E-03	4.7E-03	1.2E-03	8.7E-03	4.6E-03	2	1	yes
algz	5.5E-05	3.4E-17	3.9E-05	2.6E-03	1.9E-05	2	3	yes
ampr	8.9E-03	6.5E-03	1.2E-03	8.3E-03	4.6E-03	3	1	yes
anr	1.8E-02	5.2E-03	2.3E-02	1.0E-02	3.8E-03	3	2	yes
argr	9.4E-03	3.9E-03	1.1E-03	9.4E-03	3.9E-03	1	1	yes
bexr	9.9E-03	5.3E-03	2.3E-03	9.9E-03	5.3E-03	1	1	yes
bqsr	7.6E-03	3.7E-03	1.8E-18	7.6E-03	3.7E-03	1	1	yes
bqss	9.0E-03	4.4E-03	2.3E-03	9.0E-03	4.4E-03	1	1	yes
cbrb	8.5E-03	4.6E-03	1.1E-03	8.5E-03	4.6E-03	1	1	yes
cysb	1.6E-03	4.3E-34	1.6E-03	1.6E-03	4.3E-34	1	1	yes
dnr	7.6E-03	4.2E-03	1.8E-03	1.9E-02	6.4E-03	3	3	yes
exsa	4.8E-03	8.1E-17	5.8E-03	8.8E-03	2.4E-04	2	5	yes
exsd	8.3E-03	1.2E-17	7.1E-03	7.7E-03	6.7E-04	1	1	no
fhpr	4.8E-04	7.8E-22	3.3E-05	1.2E-03	5.9E-04	1	3	yes
fleq	6.4E-02	3.9E-17	6.8E-02	7.9E-03	3.4E-03	1	4	no
flgm	3.0E-03	7.0E-21	3.0E-03	3.9E-03	1.7E-20	1	1	no
flia	7.5E-03	4.5E-21	7.5E-03	6.6E-03	3.2E-20	1	1	no
fpvi	0.0E+00	2.0E-21	0.0E+00	3.1E-02	4.1E-03	0	1	no
fpvr	2.1E-03	4.8E-20	3.4E-03	2.9E-02	4.1E-03	2	1	no
fur	2.5E-02	1.3E-17	2.6E-02	1.7E-02	3.7E-03	5	1	no
gaca	7.4E-03	3.2E-03	1.4E-06	7.4E-03	3.2E-03	2	1	yes
gacs	8.7E-03	3.7E-03	1.1E-03	8.7E-03	3.7E-03	1	1	yes
gbur	2.3E-03	2.7E-17	9.2E-03	1.8E-02	4.1E-03	1	1	no
glpr	2.1E-03	1.2E-19	1.8E-11	2.0E-02	4.1E-03	1	1	no
gpur	9.2E-03	4.6E-03	1.1E-03	9.2E-03	4.6E-03	1	1	yes
hu	8.2E-03	5.3E-03	1.2E-03	8.0E-03	5.1E-03	2	1	yes
ihf	8.5E-03	3.2E-03	9.2E-06	8.5E-03	3.2E-03	2	1	yes
lasr	1.3E-02	5.6E-04	1.1E-02	1.3E-03	1.5E-03	6	6	no
metr	8.7E-03	5.1E-03	1.1E-03	8.7E-03	5.1E-03	1	1	yes
mexr	1.9E-02	2.0E-03	1.7E-02	1.9E-03	1.9E-03	2	1	no
mext	1.1E-02	5.9E-04	9.7E-03	1.9E-02	3.0E-03	2	2	yes
mexz	0.0E+00	3.8E-21	0.0E+00	9.0E-03	4.1E-03	0	1	no
muca	9.3E-05	2.4E-17	1.3E-04	5.8E-04	8.0E-05	1	1	no
mucb	2.7E-04	3.1E-17	2.6E-06	5.8E-04	8.0E-05	1	1	no
mucc	9.9E-03	2.1E-03	2.9E-03	9.0E-03	2.1E-03	2	1	yes
mucd	1.0E-02	4.5E-03	1.4E-03	9.7E-03	4.4E-03	2	1	yes
mvat	8.0E-03	3.7E-03	2.3E-03	8.0E-03	3.7E-03	1	1	yes
mvfr	4.0E-08	1.2E-21	8.5E-08	2.5E-03	2.7E-04	1	3	yes
mvta	9.3E-03	5.9E-03	1.3E-03	9.2E-03	5.7E-03	2	1	yes
nalc	0.0E+00	2.7E-20	0.0E+00	1.3E-02	2.0E-03	0	1	no

narl	0.0E+00	2.7E-21	0.0E+00	1.0E-02	3.8E-03	0	2	no
nfxb	1.6E-03	4.4E-34	1.6E-03	1.6E-03	4.4E-34	1	1	yes
ospr	8.7E-03	3.4E-03	2.8E-19	8.7E-03	3.4E-03	1	1	yes
pa0779	9.2E-03	3.9E-03	1.7E-03	8.5E-03	3.4E-03	2	1	yes
pa3697	8.3E-03	5.3E-03	2.3E-20	8.3E-03	5.3E-03	1	1	yes
pa5471	1.8E-02	8.3E-03	3.4E-03	9.0E-03	4.1E-03	2	1	yes
pchr	9.5E-04	6.1E-26	3.3E-04	5.5E-03	7.8E-04	1	2	yes
pfer	0.0E+00	6.7E-21	0.0E+00	1.8E-02	4.1E-03	0	1	no
phhr	1.6E-03	3.9E-34	1.6E-03	1.6E-03	3.9E-34	1	1	yes
phop	2.6E-02	1.8E-02	8.0E-03	8.7E-03	5.3E-03	3	1	yes
phoq	2.5E-02	6.5E-04	1.6E-02	8.7E-03	5.3E-03	2	1	no
pila	4.5E-03	1.0E-22	4.7E-03	4.9E-03	1.9E-04	1	3	yes
pilr	8.5E-03	4.7E-03	8.8E-06	8.3E-03	4.6E-03	2	1	yes
pmra	3.9E-03	2.8E-03	1.1E-03	1.3E-02	3.9E-03	2	2	yes
pprb	9.0E-05	8.5E-19	1.0E-04	9.9E-03	3.5E-08	2	2	no
ppyr	9.4E-03	4.1E-03	2.4E-19	9.4E-03	4.1E-03	1	1	yes
pqrr	1.0E-02	5.1E-03	3.4E-03	1.0E-02	5.1E-03	1	1	yes
psdr	9.9E-03	3.9E-03	1.1E-03	9.9E-03	3.9E-03	1	1	yes
psra	1.9E-03	9.8E-21	3.1E-03	1.6E-03	4.1E-34	3	1	yes
ptxr	5.3E-05	2.2E-22	2.1E-05	5.7E-03	6.8E-04	1	4	no
ptxs	9.4E-03	5.0E-23	8.0E-03	5.1E-03	7.8E-05	2	4	yes
pvds	1.0E-04	1.0E-19	7.9E-05	2.1E-03	5.8E-08	2	3	no
qscr	8.3E-03	5.3E-03	1.1E-03	8.3E-03	5.3E-03	2	1	yes
rcdb	8.5E-03	5.1E-03	1.6E-19	8.5E-03	5.1E-03	1	1	yes
rhlr	3.3E-04	3.0E-17	5.8E-05	6.5E-04	2.8E-04	3	7	yes
roca1	8.5E-03	3.4E-03	1.1E-03	8.5E-03	3.4E-03	1	1	yes
rocr	9.7E-03	4.6E-03	7.3E-20	9.7E-03	4.6E-03	1	1	yes
roxr	8.7E-03	4.6E-03	5.9E-19	8.7E-03	4.6E-03	1	1	yes
roxs	8.5E-03	4.6E-03	1.1E-03	8.5E-03	4.6E-03	1	1	yes
rpod	5.9E-03	6.8E-18	2.0E-04	9.0E-03	4.1E-03	2	1	no
rpon	8.3E-03	4.7E-03	1.2E-03	8.0E-03	4.4E-03	4	1	yes
rpos	2.6E-04	2.4E-17	3.4E-04	2.9E-03	2.8E-03	1	4	no
rsal	0.0E+00	3.6E-21	0.0E+00	3.3E-03	2.0E-04	0	2	no
toxr	0.0E+00	2.0E-20	0.0E+00	2.1E-03	1.3E-09	0	1	no
tpba	3.5E-03	3.5E-23	3.3E-03	4.3E-03	4.8E-04	1	3	yes
trpi	1.6E-03	4.6E-34	1.6E-03	1.6E-03	4.6E-34	1	1	yes
vfr	5.9E-02	2.7E-02	1.0E-02	9.4E-03	3.7E-03	5	1	yes
vqsm	1.5E-02	9.9E-03	7.8E-04	9.7E-03	4.1E-03	4	1	yes
vqsr	9.0E-03	2.1E-17	6.9E-03	3.0E-03	3.6E-03	1	2	no
vrei	9.2E-03	3.0E-03	2.3E-03	9.2E-03	3.0E-03	1	1	yes

7.4.10 E. coli

Node	**TC**	**VC**	**DC**	**TC-Sens.**	**VC-Sens.**	**Outgoing**	**Incoming**	**Loop**
acrr	9.9E-04	2.4E-34	9.9E-04	9.9E-04	2.4E-34	1	1	yes
ada	1.5E-03	6.6E-22	1.6E-03	1.3E-18	3.0E-20	3	2	yes
adiy	1.0E-07	6.3E-25	3.2E-08	1.7E-03	2.5E-03	1	1	no
agar	9.9E-04	3.1E-34	9.9E-04	9.9E-04	3.1E-34	1	1	yes
aidb	1.7E-21	8.1E-26	1.9E-25	1.5E-03	3.2E-20	1	3	yes
alas	9.9E-04	1.7E-34	9.9E-04	9.9E-04	1.7E-34	1	1	yes
allr	4.7E-03	2.3E-03	4.8E-23	4.7E-03	2.3E-03	1	1	yes
alsr	5.2E-03	2.8E-03	1.4E-03	5.2E-03	2.8E-03	1	1	yes
arac	5.4E-22	3.5E-20	2.2E-22	3.5E-18	4.2E-20	2	3	yes
arca	2.5E-04	8.5E-17	6.9E-04	1.6E-05	5.1E-06	7	2	no
argp	2.2E-03	8.3E-25	2.3E-03	1.9E-03	3.2E-04	2	2	yes
argr	9.9E-04	2.4E-34	9.9E-04	9.9E-04	2.4E-34	1	1	yes
arsr	9.9E-04	2.6E-34	9.9E-04	9.9E-04	2.6E-34	1	1	yes

ascg	5.9E-03	3.0E-03	6.9E-04	5.9E-03	3.0E-03	2	1	yes
asnc	8.7E-04	0.0E+00	1.0E-03	1.1E-03	1.5E-04	1	2	yes
baer	8.8E-04	5.5E-04	9.0E-25	2.9E-03	1.3E-03	1	2	yes
basr	4.5E-03	2.8E-03	2.2E-06	4.4E-03	2.8E-03	3	1	yes
beti	4.4E-04	9.9E-26	1.9E-04	2.1E-03	5.1E-04	1	3	yes
cadc	8.5E-07	1.8E-24	2.7E-06	3.0E-03	8.9E-04	1	5	yes
cdar	4.8E-03	2.3E-03	2.1E-23	4.8E-03	2.3E-03	1	1	yes
chbr	0.0E+00	2.8E-25	0.0E+00	9.9E-04	2.4E-22	0	2	no
cpxr	8.0E-03	3.1E-03	2.1E-03	5.2E-03	1.8E-03	3	1	yes
cra	1.4E-03	1.5E-23	1.2E-03	1.8E-03	2.2E-03	7	1	no
crp	2.0E-02	3.0E-03	1.4E-02	2.3E-03	7.9E-22	41	3	yes
csgd	3.9E-08	1.1E-24	4.6E-08	1.7E-03	1.3E-03	1	12	yes
cspa	5.7E-03	2.5E-03	7.7E-23	5.7E-03	2.5E-03	1	1	yes
cusr	9.9E-04	5.5E-04	4.6E-22	2.5E-03	1.1E-03	1	2	yes
cynr	9.9E-04	2.3E-34	9.9E-04	9.9E-04	2.3E-34	1	1	yes
cysb	2.1E-24	9.5E-25	6.6E-24	9.9E-04	1.2E-19	1	2	yes
cytr	9.9E-04	3.0E-34	9.9E-04	9.9E-04	3.0E-34	1	1	yes
dan	5.5E-03	2.2E-03	6.9E-04	5.5E-03	2.2E-03	1	1	yes
dcur	2.1E-05	7.9E-21	3.1E-06	1.4E-03	4.5E-04	1	3	no
dhar	9.9E-04	1.9E-34	9.9E-04	9.9E-04	1.9E-34	1	1	yes
dnaa	1.0E-03	9.9E-27	1.1E-03	1.6E-03	9.0E-05	1	2	yes
dpia	5.7E-08	3.3E-20	8.6E-08	8.2E-05	2.4E-05	1	6	yes
dsdc	2.9E-04	4.4E-28	3.6E-04	9.3E-04	4.9E-04	1	2	yes
evga	5.7E-04	5.7E-04	8.7E-07	4.8E-03	2.6E-03	3	2	yes
exur	9.9E-04	4.7E-28	9.9E-04	9.9E-04	3.1E-34	2	1	yes
fadr	5.6E-03	2.8E-03	9.9E-04	5.4E-03	2.5E-03	2	1	yes
fhla	5.5E-04	7.1E-18	4.6E-06	3.9E-03	2.3E-03	2	4	yes
fis	2.1E-03	1.1E-04	2.3E-03	2.4E-03	1.9E-04	9	3	yes
flhdc	5.7E-03	2.5E-03	4.6E-06	5.7E-03	2.5E-03	2	1	yes
fliz	6.6E-03	8.0E-05	6.8E-03	1.8E-03	1.6E-03	3	4	no
fnr	1.2E-02	8.1E-05	1.0E-02	1.5E-03	6.1E-04	18	4	yes
fucr	4.9E-03	7.7E-04	3.9E-03	6.1E-03	2.4E-03	1	2	yes
fur	8.4E-03	6.3E-19	8.4E-03	2.0E-04	1.3E-05	7	4	yes
gade	2.6E-04	4.8E-20	2.6E-04	5.0E-03	2.5E-03	4	9	yes
gade-rcsb	5.4E-03	2.6E-03	6.7E-09	5.4E-03	2.6E-03	2	1	yes
gadw	3.1E-05	1.6E-20	4.3E-07	1.5E-04	1.1E-04	3	10	yes
gadx	3.7E-07	1.9E-21	1.3E-06	5.0E-05	2.0E-06	4	14	yes
galr	1.7E-03	5.0E-27	1.7E-03	2.1E-03	2.2E-22	2	2	yes
gals	1.0E-04	9.3E-25	1.0E-04	6.8E-04	2.5E-22	2	3	yes
gcva	9.9E-04	2.3E-34	9.9E-04	9.9E-04	2.3E-34	1	1	yes
glcc	3.2E-25	7.1E-23	1.1E-25	8.5E-04	7.0E-22	1	4	yes
gntr	9.9E-03	2.6E-03	3.3E-03	5.5E-03	1.8E-03	2	1	yes
gutm	4.3E-06	7.7E-20	3.7E-26	4.9E-03	1.6E-03	1	4	yes
gutr	1.0E-02	3.1E-03	2.7E-03	5.8E-03	1.8E-03	2	1	yes
hcar	9.9E-04	2.1E-34	9.9E-04	9.9E-04	2.1E-34	1	1	yes
hdfr	0.0E+00	2.8E-24	0.0E+00	1.7E-03	2.5E-03	0	1	no
hipab	5.6E-03	3.9E-03	1.6E-03	5.1E-03	3.3E-03	2	1	yes
hipb	2.5E-04	1.3E-23	1.8E-04	8.1E-04	7.2E-04	1	3	yes
h-ns	3.1E-02	2.0E-02	2.5E-02	5.2E-03	2.5E-03	17	1	yes
hyfr	1.9E-07	5.6E-19	1.9E-07	2.8E-03	5.1E-06	1	4	yes
iclr	2.2E-03	6.1E-26	1.9E-03	2.5E-03	3.6E-04	1	2	yes
idnr	6.6E-04	3.3E-04	3.0E-22	5.5E-03	1.5E-03	1	3	yes
ihf	9.4E-03	6.9E-03	1.9E-03	5.5E-03	2.2E-03	10	1	yes
ilvy	9.9E-04	2.2E-34	9.9E-04	9.9E-04	2.2E-34	1	1	yes
iscr	9.9E-04	3.1E-34	9.9E-04	9.9E-04	3.1E-34	1	1	yes
leuo	7.4E-04	8.8E-21	1.2E-05	5.3E-03	2.3E-03	2	4	yes
lexa	9.9E-04	1.8E-34	9.9E-04	9.9E-04	1.8E-34	1	1	yes
lldr	1.8E-03	0.0E+00	1.8E-03	1.9E-03	2.0E-06	1	2	yes
lrha	5.4E-03	1.7E-03	6.9E-04	5.4E-03	1.7E-03	1	1	yes

lrp	1.3E-03	9.1E-18	1.1E-03	6.9E-04	2.7E-04	4	4	yes
lsrr	2.3E-24	2.4E-25	2.0E-24	9.9E-04	9.2E-22	1	2	yes
lysr	9.9E-04	2.2E-34	9.9E-04	9.9E-04	2.2E-34	1	1	yes
mali	1.4E-24	2.7E-25	1.3E-26	9.9E-04	8.5E-22	1	3	yes
mara	2.9E-04	1.1E-23	7.6E-05	2.1E-03	1.3E-04	4	7	yes
marr	3.9E-03	1.9E-24	5.1E-03	2.1E-03	1.3E-04	2	7	yes
mata	7.8E-03	4.9E-03	2.1E-03	5.7E-03	3.0E-03	2	1	yes
maze	5.0E-04	2.5E-26	6.3E-04	8.7E-04	2.1E-04	1	5	yes
maze-mazf	6.4E-03	2.9E-03	8.8E-04	6.1E-03	2.8E-03	2	1	yes
melr	1.2E-21	3.6E-23	8.7E-22	4.6E-18	6.0E-21	2	3	yes
metj	7.1E-04	7.7E-25	7.1E-04	6.9E-03	1.2E-06	1	1	no
metr	2.3E-05	3.2E-30	2.3E-05	2.1E-03	3.8E-08	1	2	yes
mlc	3.5E-24	1.0E-24	1.4E-26	9.9E-04	7.4E-22	1	3	yes
mlra	6.0E-06	4.6E-23	6.8E-04	8.6E-03	1.9E-03	2	1	no
mngr	9.9E-04	2.4E-34	9.9E-04	9.9E-04	2.4E-34	1	1	yes
mode	5.5E-03	3.8E-03	1.9E-03	4.8E-03	2.9E-03	2	1	yes
mpra	9.9E-04	2.6E-34	9.9E-04	9.9E-04	2.6E-34	1	1	yes
mtlr	6.0E-26	3.5E-25	2.8E-25	8.5E-04	6.6E-22	1	4	yes
murr	4.8E-03	2.9E-03	7.5E-23	4.8E-03	2.9E-03	1	1	yes
nac	4.3E-04	1.8E-26	6.4E-04	7.7E-04	4.7E-04	2	4	yes
nagc	9.9E-04	1.4E-26	9.9E-04	9.9E-04	2.4E-34	2	1	yes
narl	3.2E-04	4.5E-17	4.3E-04	4.4E-03	2.0E-03	3	2	no
nemr	7.0E-05	8.2E-34	7.0E-05	1.4E-04	1.9E-33	1	2	yes
nhar	2.8E-04	6.9E-04	8.4E-25	5.2E-03	2.2E-03	1	2	yes
nikr	7.2E-04	7.7E-26	1.2E-03	1.2E-03	1.9E-04	1	3	yes
norr	3.6E-04	1.0E-29	2.2E-05	1.0E-03	4.9E-04	1	2	yes
nsrr	1.3E-02	7.9E-03	6.6E-03	6.1E-03	2.5E-03	6	1	yes
ntrc	5.8E-03	3.0E-03	8.3E-04	5.5E-03	2.5E-03	2	1	yes
ompr	1.3E-07	2.9E-25	2.0E-07	1.5E-03	6.7E-22	1	3	no
oxyr	1.5E-06	7.7E-17	1.9E-06	9.9E-04	7.2E-22	2	2	yes
pdhr	4.8E-24	6.2E-24	5.3E-24	1.2E-03	7.1E-22	1	3	yes
pepa	9.9E-04	2.4E-34	9.9E-04	9.9E-04	2.4E-34	1	1	yes
phob	9.4E-03	6.2E-03	8.2E-04	5.1E-03	2.2E-03	7	1	yes
phop	9.9E-04	3.4E-19	9.9E-04	2.7E-18	1.2E-19	7	2	yes
prpr	4.5E-24	2.7E-25	8.7E-25	1.3E-04	2.2E-22	1	4	yes
pspf	9.9E-04	2.3E-34	9.9E-04	9.9E-04	2.3E-34	1	1	yes
purr	1.8E-03	0.0E+00	1.8E-03	1.9E-03	5.1E-06	1	2	yes
puta	1.5E-05	1.2E-25	5.3E-05	4.5E-04	5.5E-05	1	3	yes
puur	1.4E-03	0.0E+00	1.4E-03	1.5E-03	1.7E-06	1	3	yes
qseb	5.9E-03	1.9E-03	6.9E-04	5.9E-03	1.9E-03	1	1	yes
rbsr	7.9E-25	2.9E-25	6.9E-24	9.9E-04	7.9E-22	1	2	yes
rcda	4.1E-03	3.5E-03	6.9E-04	4.1E-03	3.4E-03	2	1	yes
rcnr	9.9E-04	2.7E-34	9.9E-04	9.9E-04	2.7E-34	1	1	yes
rcsab	5.4E-03	3.2E-03	7.1E-04	4.6E-03	2.9E-03	2	1	yes
rcsb	5.1E-04	2.0E-17	5.7E-04	1.7E-03	2.5E-03	2	1	no
rcsb-bglj	6.3E-03	3.3E-03	2.9E-06	4.6E-03	2.2E-03	2	1	yes
relb	6.0E-04	1.7E-26	1.0E-03	1.1E-03	4.6E-04	1	2	yes
relb-rele	5.5E-03	2.8E-03	4.1E-04	5.0E-03	2.3E-03	2	1	yes
rhar	1.7E-03	6.3E-24	1.1E-04	2.1E-03	1.7E-21	2	3	yes
rhas	2.0E-03	2.9E-23	2.8E-03	2.1E-03	2.0E-21	2	3	yes
rob	1.8E-03	3.5E-27	2.0E-03	2.2E-03	6.4E-05	3	3	yes
rsta	3.5E-06	1.7E-24	3.0E-06	3.6E-18	1.5E-19	1	1	no
rutr	1.1E-03	3.6E-19	1.1E-03	9.9E-04	3.5E-34	4	1	yes
sdia	5.0E-03	2.2E-03	6.9E-04	5.0E-03	2.2E-03	2	1	yes
sgrr	9.9E-04	3.0E-34	9.9E-04	9.9E-04	3.0E-34	1	1	yes
soxr	8.7E-04	7.1E-20	7.8E-04	6.6E-04	1.2E-04	2	3	yes
soxs	6.5E-04	5.9E-17	8.1E-04	1.2E-03	1.5E-04	5	4	yes
stpa	4.9E-04	6.3E-22	1.2E-03	7.4E-04	1.1E-04	2	3	yes
tdca	1.9E-05	3.0E-24	6.6E-05	8.9E-04	1.0E-03	1	5	yes

tdcr	5.2E-03	3.5E-03	1.3E-03	4.6E-03	3.0E-03	2	1	yes
torr	9.9E-04	9.9E-23	9.9E-04	9.9E-04	2.4E-34	2	1	yes
trpr	9.9E-04	2.8E-34	9.9E-04	9.9E-04	2.8E-34	1	1	yes
tyrr	9.9E-04	2.6E-34	9.9E-04	9.9E-04	2.6E-34	1	1	yes
uidr	9.9E-04	2.8E-34	9.9E-04	9.9E-04	2.8E-34	1	1	yes
uxur	2.9E-22	4.3E-25	6.2E-24	5.3E-04	2.9E-22	1	3	yes
xylr	2.4E-10	5.7E-21	3.1E-10	7.7E-04	1.8E-08	1	4	yes
ydeo	7.9E-04	1.4E-22	2.7E-04	1.3E-03	7.2E-04	2	5	yes
yefm	4.0E-04	7.0E-30	3.6E-04	1.0E-03	3.8E-04	1	2	yes
yefm-yoeb	6.6E-03	2.3E-03	1.8E-03	5.9E-03	1.9E-03	2	1	yes
yeil	1.8E-04	8.6E-20	9.5E-07	5.5E-03	6.2E-04	1	4	yes
yiaj	3.3E-04	6.6E-32	1.2E-05	1.2E-03	4.4E-04	1	3	yes
yqji	1.8E-03	1.9E-26	1.3E-03	2.3E-03	1.6E-04	1	2	yes
zrar	5.1E-03	9.9E-04	3.3E-03	6.1E-03	2.0E-03	1	2	yes

7.5 Number of iterations in the approximation algorithms

In Jimena, the integrals and the minimum in the calculation of the dynamic centrality are approximated numerically by random sampling with different numbers of samples:

	TC/VC	**DC (integral)**	**DC (minimum)**
Mouse colon subnet	100	-	-
Human T-helper differentiation	1000	100	50
Mammalian chondrocyte regulation	1000	80	50
A. thaliana inflorescence	1000	100	50
A. thaliana immune response	100	20	20
A. thaliana root stem cell niche	1000	150	80
S. pombe (fission yeast) cell cycle	100	100	50
S. cerevisiae (budding yeast)	50	10	10
P. aeruginosa	50	10	10
E. coli	50	10	10

Acknowledgements

First and foremost, this thesis would not have been possible without the tireless support of Prof. Thomas Dandekar, who gave me the freedom to pursue my own ideas, and was an inexhaustible source of invaluable discussions, comments and encouragements.

I would also like to extend my sincere gratitude to

- Martin Kaltdorf and Chunguang Liang for helpful suggestions regarding Jimena.
- Jennifer Heilig for stylistic comments and native speaker corrections on (Karl, Dandekar 2015).

Mere words cannot express the gratitude towards my parents who supported my studies and this thesis in every way possible.

This research was supported by Deutsche Forschungsgemeinschaft grants Da 208/12-1 and TR 124/B1.