



Project no. FP6-NEST-2003-1-12789
ESIGNET
Evolving Cell Signalling Networks in Silico

Specific Targeted Research Project
Sixth Framework Programme Priority

Deliverable number 4.2
Specification of the Properties of Cell Signalling Systems
Document Describing Framework to Specify Desired Computational Properties

Due date of deliverable: May 2006
Actual submission date: May 2006

Start date of project: 2005-09-01

Duration: 36 months

Friedrich Schiller University Jena

Revision: final

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Abstract

The overall goal of the ESIGNET project is to study the computational properties of cell signalling networks (CSN) by evolving them using methods from evolutionary computation, and to re-apply this understanding in developing new ways to model and predict real CSNs.

Finding an appropriate framework to specify the desired computational properties of CSNs is essential for all subsequent parts of the ESIGNET project. This includes identification of objectives with regard to the target solution which is represented as a formal description of CSNs. In this report, we summarise different criteria important for CSN representation, simulation, evaluation, and optimisation. These criteria as a whole form a framework oriented to an effective and efficient analysis *in silico*. We distinguish between criteria containing descriptive, structural, dynamical, and functional properties. Beyond these criteria, we introduce different levels for formal representation of target functions able to specify CSN behaviour from a computational point of view. In addition to this, selected techniques useful for CSN specification and evaluation are presented.

Contents

1	Introduction	2
2	Descriptive Properties	3
3	Structural Properties	4
4	Dynamical Properties	5
5	Functional Properties	6
6	Specification of CSN Target Functions	7
7	Aspects of CSN Fitness Measure and Evaluation	10
8	Conclusions	11

1 Introduction

The ESIGNET workpackage (WP) 4 includes research activities leading to formal specifications of CSNs and their evaluable properties. In fulfilment of this, WP4 is thought to focus on following steps:

- Develop suitable ways to describe the computational properties of CSNs.
- Define phenotypic representations of a CSN and its dynamical behaviour.
- Implement the phenotypic representation.
- Find a suitable measure for distance between the computational properties of two CSNs.
- Explore fast methods to derive a fitness function given a description of a CSN and the specification of a target CSN.
- Development of user interface software for specification of CSN properties

Objectives within WP4 are specified by five deliverables:

- D4.1.** Document describing the formats for the phenotypic representation of CSNs and their dynamical behaviour.
- D4.2.** Document describing the framework to specify the desired computational properties of the target solution.
- D4.3.** Software that performs simulation, reads specification of CSN properties, and calculates fitness for a given CSN and a target CSN.
- D4.4.** Publication in peer-reviewed journal and/or conference.
- D4.5.** Software to automate the input of the specification of target CSNs.

Modelling phenomena from nature is a basic motivation for development and refinement of mathematics. Beside traditional natural, engineering, and social sciences, modern systems biology requires exhaustive application of mathematical principles. Success in genomics and proteomics can discover interconnections between structure of biomolecules and function of biological systems formed by those biomolecules. CSNs can be seen as a special class of biological systems consisting of interacting proteins and auxiliary substances for the purpose of organizing information processing inside living organisms. Identifying what kind of information is encoded by cell signals and finding the way how cell signals are generated, transferred, modified, and utilised should be reflected by mathematical models of CSNs. Furthermore, formal descriptions define a certain level of abstraction. High abstraction levels are suitable for recognition of general conclusions, but they weaken the view into details of system components. Low abstraction levels retrieve details about selected system components, but they produce a huge amount of low structured data. We intend to serve both by a variety of different approaches to model CSNs.

From a pragmatic point of view, four classes of CSN properties can be distinguished. Concerning their occurrence in the process of generation, modification, and evaluation, they form a classification. It consists of descriptive, structural, dynamical, and functional properties, in this sequence with increasing complexity. They contain partially overlapping as well as contradicting

criteria. Balancing and weighting between these criteria can be seen as one of the challenges of the ESIGNET project. Since this report should correspond to deliverable D4.2, we provide an annotated list of criteria, point out their consequences and make suggestions how to measure them. The list gives a framework for subsequent workpackages.

2 Descriptive Properties

Descriptive properties define basic requirements to the formalism used to denote CSNs. Easy to check, descriptive properties facilitate the generation of promising network candidates ab initio and help to avoid nonsense networks in advance. Furthermore, seminal decisions regarding purpose and intention for ongoing studies are reflected in descriptive properties. For more detailed information, see [1, 4, 9].

Representation: There exists a plethora of formalisms to represent CSNs in terms of formal systems. That also implies their behaviour and capabilities. For an overview, see the ESIGNET report regarding deliverable D4.1. The choice of the mathematical instrument used for modelling is essential for the kind of information and cognitions to achieve. In principle, analytical, stochastic, algebraic, and category based approaches are available.

Finiteness: Each network description should be composed of finite elements that are combined within a finite system. Otherwise, the network is difficult to process in silico, algorithms for evaluative analysis can fail or produce restricted results. Finite networks correspond to nature. Cardinalities of system components and configuration (state) space allow to measure (quantify) finiteness.

Completeness: All necessary elements of the system description should be present, all required external parameters should be initialised with values. There are no missing or unspecified components. Completeness is a prerequisite for formal verification and validation. Its presence or absence can be used as a weighted factor within CSN fitness evaluation.

Correctness: All elements of the system description should meet the underlying assertions and constraints. The values of external parameters are allowed to vary within the predefined range. For instance, this means in terms of network topology that all edges are assigned to nodes. Correctness with regard to a formal specification can be proven by using reasoning strategies based on soundness and completeness. Presence or absence of proveable correctness can be used for CSN fitness evaluation.

Computational Tractability: The possibility to apply simulation or continuative analysis methods is expressed by the term computational tractability. This depends on the used formalism for system description in comparison to the methods for analysis. Questions of decidability and determinism play a key role here. Computational tractability is essential for formal description of CSN behaviour, comparison to target functions and fitness evaluation.

Abstractness: The level of abstraction defines the view onto the system. The decision about what kind of information is considered and what kind of data is ignored spans the range for the achievable knowledge. Beyond microscopic and macroscopic approaches, quantity and structural degree of result data are determined. CSN comparison and evaluation issues require equal levels of abstraction.

Granularity: The choice of system elements and the intensity of their interconnection specifies a certain granularity of the system description. As an example, pools of biomolecules can be modeled as finite sets of objects or using concentrations. Finite sets of objects allow a molecular handling in a one-to-one manner, while concentrations refer to statistical summaries. CSN comparison and evaluation issues require similar granularity to achieve reasonable results.

Complexity: Cardinalities of system elements, the degree of nesting, and restrictions of permitted compositions for system internal rules belong to complexity issues. System theory and complexity theory distinguish complexity classes and their consequences to subsequent studies. Complexity measures quantify properties of CSN structure.

3 Structural Properties

Structural properties are derived from a static system description with fixed components. A bit more specific than descriptive properties in general, they are suitable for a first stage of fitness evaluation without any need of extensive dynamical studies. For more detailed information, see [3, 10, 11].

Topology: With respect to networks, topology marks one of the basic properties. Quantitative items like size or weight are good candidates for optimisation targeting. Qualitative items like occurrence of feedbacks, accessibility, circular or nondisjoint structures classify the network.

Physical and Chemical Plausibility: This property subsumes all considerable constraints coming from natural laws. Examples are beside mass-conservance also general thermodynamic limitations like energy conservation, minimisation of free energy or increase of entropy.

Robustness to Structural Perturbations: With respect to the biological template of CSNs, network edges can model pathways or pathway subunits. The question might be of interest, whether or not removed/blocked pathways can be replaced by others keeping accessibility. Structural robustness qualifies a network to keep its function in case of modifications. This requires a certain redundancy within the network topology. Critical path analysis, network flow analysis, spanning tree analysis, and exhaustive testing of CSNs with random structural perturbations allows to measure this property.

Modularity: Engineering design principles often produce network topologies consisting of modular subunits. Subunits are connected using well-defined interfaces. Modularity defines a structural property that allows to create more complex network systems by composition and arrangement of repetitive subunits. For instance, building blocks, receptors or cytosoles can act as subunits. A typical feature of subunits is their re-use. The degree of modularity, percentage of re-used subunits, and intensity of data exchange between subunits gives measurable results here.

Ability to Identify Objects / Substructures: In contrast to modularity, the analysis of unknown networks intends to identify subunits (modules) inside a complex structure. Methods taken

from organisation theory can be helpful to detect modules. Furthermore, the huge knowledge pool provided by databases of bioinformatics enables support in comparison and homology search. Degrees of similarity between identified subunits and those integrated in molecular biological databases help to obtain evaluable measures for this property.

4 Dynamical Properties

Dynamical properties become visible after observation of the system behaviour for a certain amount of time. This can include infinite time spans. Time- or frequency-based analysis often consumes a large amount of computing resources for numeric simulations. Therefore, dynamical properties facilitate a much more detailed view into a system than static ones. Forming the second stage of fitness evaluation, methods to deliver dynamical properties differ with respect to their time-discrete or time-continuous nature. For more detailed information, see [6, 7, 10, 12].

Parameterisation: The amount, definition, granularity, and domain of parameters involved in the extrapolation of the system status forms the basic dynamic property. Parameters express the flexibility of a system to react on changes of any kind. Identification of suitable evolutionary parameters can be seen as a substantial part of the ESIGNET project.

Need of Resources: Processes inside CSNs require consumption of resources. The number of dynamically acting system components determines the amount of resources. Thus, minimisation of the need of resources has high priority in terms of system optimisation. Since resources are always limited, economic issues are essential for the survival in nature. Questions about shared use of components like crosstalk could also be tackled here.

Confluence: Coming from theoretical computer science, this property expresses the ability of a system to achieve same results from same input in different ways. For computational reasons, confluence is necessary to guarantee correctness. Nondeterministic behaviour can be systematised and optimised by understanding confluency. Confluency is a prerequisite for re-usage in terms of computational devices. Confluency analysis based on algebraic approaches decides about presence or absence of this property for CSN fitness evaluation.

Stability: Small changes of input data should keep the system behaviour in principle. The dynamical stability of system behaviour decides about its practicability and about its quality to repeat processes. Radius and speed of convergence can measure this kind of stability. In contrast, numeric stability characterises a quality of applied algorithms.

Asymptotic Behaviour: Assumed system observation for an infinite amount of time classifies the asymptotic behaviour. From this point of view, system components can converge, diverge, oscillate or become undefined. Convergence leads to a steady state, Oscillation occurs within fixed point orbitals.

Monitoring: Simulating dynamical behaviour in slow/fast motion or from an unusual perspective can emphasise desired aspects of a system. Systems equipped with monitoring features can be analysed in manifold way and even by unconventional methods. Starting from computer animation and not limited to data mining strategies, monitoring could be a clue for getting promising inspirations. An example is the identification of key spots in the evolutionary development process. Furthermore, statistical and classification methods on

monitoring data can answer questions about applicability, adaptability, and suitability of CSN candidates.

5 Functional Properties

The most complex properties of CSNs as a target solution concern functional issues. Beyond structural, static, and dynamical analysis, a wide understanding of the considered system is a prerequisite. The interdisciplinary nature of the ESIGNET project becomes most obvious in answering questions about functional properties of biological systems. Bioinformatics and especially proteomics aim to discover relations between structure, behaviour, and function. The rapidly growing knowledge in this field is also an important potential for ESIGNET. The subsequent list of functional properties gives some raw ideas thought to be extended within the project. For more detailed information, see [2, 5, 8].

Encoding of Signals: The theory of information and coding provides fundamental aspects about the nature of signals and their transduction. Can we integrate cell signalling in this context? A measurable representation of signal encoding is entropy.

Precision of Signal Transduction / Computation: Signal transduction via channels can undergo external and internal influences. These influences and mechanisms of signal refreshing/amplification determine the precision of intracellular signal processing. Interpreting signal processing as a kind of computation, the quality of output data can be used to compare and to evaluate the signalling as a whole.

Speed of Signal Transduction / Computation: Identification of bottlenecks in signalling networks is an example of how information about signalling speed and fluxes can be applied. Furthermore, knowledge about this property can help to explain reaction mechanisms in living organisms.

Redundancy: Between contradicting requirements of a minimum network size and a maximum of reliability, redundancy can balance. Redundancy in this sense relates to both signal encoding and transduction. A way how to measure redundancy consists in the compression rate achievable by different compression techniques.

Robustness to Perturbations: Efficient signalling processes can be disturbed in manifold way. The occurrence of perturbations includes all parts of a signal processing system. The more robust a system is constructed the better it will stay alive. Exhaustive robustness tests using randomly generated perturbation data provide a statistical measure here.

Computational Capability: Signal processing systems form models for computation. Their computational completeness (Turing completeness) is already shown. Any terminating algorithm denoted as a universal model for computation can be transformed into a chemical reaction network. It remains open which (sub)classes of computability signalling networks reach with limited resources.

Ability of Self-Repairation, Self-Maintenance, and Self-Organisation: Self-X properties have been identified in several disciplines like organic computing. Since CSNs belong to living systems, the complete understanding of underlying mechanisms and their embedding in the whole system is a crucial aspect for decoding the secret of life.

Ability of Multi-Use: How long, how intensive, and how often a CSN can be used before apoptosis is described by the multi-use property. Answering the question, whether or not system components drift, leads to conclusions about reliability and practicability. Methods to describe drifting could be integrated into the fitness evaluation.

6 Specification of CSN Target Functions

The variety of computational properties of CSNs discussed above provides a framework for CSN description, specification of target functions, and fitness evaluation. This framework is introduced in a quite general manner since it should serve for subsequent more detailed studies within the ongoing project. Setting an initial point for these kinds of further investigations, we give some basic ideas for specification of CSN target functions in this section.

Levels of Formal Representation and Computational Power

According to computational power and complexity, different levels of representation can be distinguished to specify CSN target functions. We introduce representation strategies from simple to extensive approaches.

Input-Output-Tables

The simplest way to define a target function consists of an output-table. For a finite number of discrete time points, the expected number of individuals (e.g. objects, molecules) of a declared output species is listed. In this case, the network input is given by its initial configuration in terms of a constant. Several inputs and dedicated output-tables can be combined for one CSN. The input is handled in a static manner, it cannot be modified over time.

An extension lies in the declaration of a CSN input species which can be controlled externally. This allows modifications during computational processes in terms of responding to signals. The input species and the initial network configuration encode the whole input of the system. Predefined input-output-tables considering a finite number of discrete time points provide an opportunity that is easy to use.

Introduction of don't-care-symbols (placeholders) acting as a kind of meta symbols makes input-output-tables a more flexible tool in utilisation. Placeholders stand for arbitrary values. They do not influence the fitness evaluation. The main advantage of placeholders can be seen in the ability to define time spans wherein the network tunes to its expected behaviour.

Explicitly Defined Arithmetic Functions

Describing target functions using input-output-tables can require a lot of effort, especially in case of trajectory refinement and parameterised functions. Particularly, shifting, expanding, or smoothing of trajectories can be described effectively using parameterised functions. Replacing input-output-tables by explicitly defined arithmetic functions necessitates techniques for interpretation of function terms. Those techniques are established. Providing a set of basic functions and construction principles, a syntactical space for permitted functions is defined inductively. Additionally, the usage of arithmetic functions facilitates coping with oscillating behaviour.

Terminating Programs Denoted as Models for Computation

A more general approach utilises results of arbitrary computational processes to describe target functions of CSNs. To do so, any terminating computer program can be transferred into a target CSN with specified behaviour. Since reaction networks are known to be computationally complete (universal, Turing-complete), they are able to perform each computation that can be run on a Turing machine. There are well-known transformations between universal models of computation, see figure 1. Each transformation is mapped constructively to an algorithm. Among deterministic models for computation, these algorithms own a polynomial time complexity, while change between deterministic and nondeterministic paradigms causes exponential time complexity since the tree representing all computational paths has to be sequentialised.

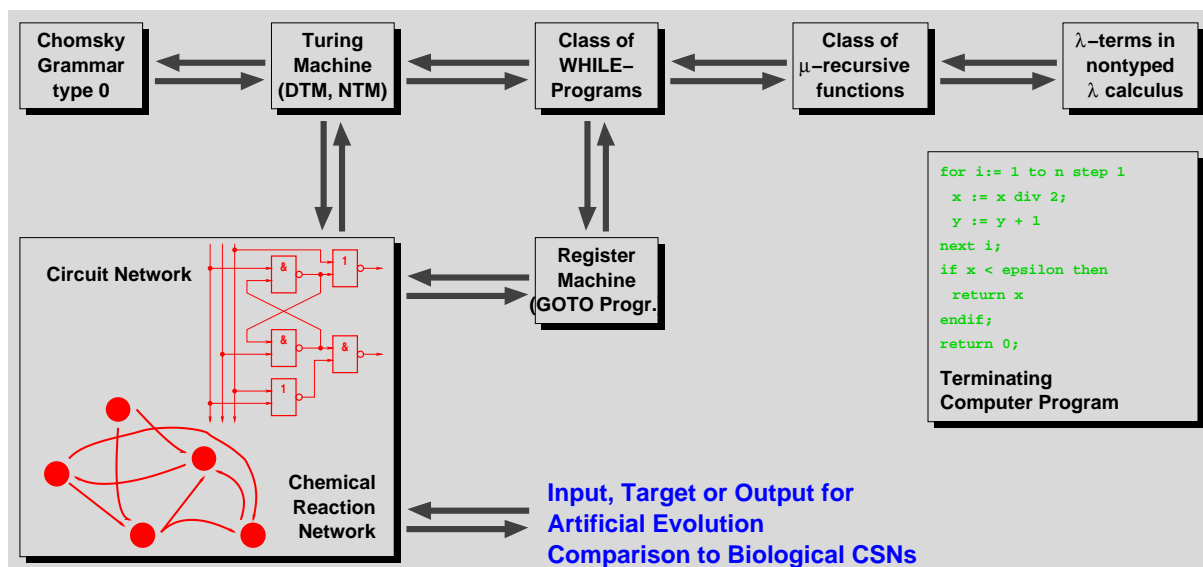


Figure 1: Transformations between some universal models of computation. All transformations are formalised by algorithms. In consequence, any terminating computation process, denoted using an arbitrary model proven to be universal, can serve as target for CSNs.

Specifications by Category Theoretical Description

Category theoretical descriptions are based on combinations of verifiable properties. Classes of objects with same combinations of properties form categories. Homomorphisms and homology mappings (functors) between them formalise degrees of similarity. Properties are denoted as logical expressions (terms) that can be inferred from axioms resulting from the basic elements the model is composed of. Category theoretical approaches offer the most general way to specify target functions and CSNs directly by favoured properties. Contradicting combinations of properties lead to empty classes of fulfilling target functions or CSNs. A challenge of category theoretical descriptions consists in coping with infinite numbers and uncertainties of instances belonging to one class. Furthermore, proving soundness and completeness requires a logical formalisation of all properties used within descriptions.

Selected Techniques to Facilitate Specifications

Target functions represent the expected behaviour of a CSN. The behaviour concerns static (asymptotic) as well as dynamical aspects. Coupled to absolute molecule numbers or concentrations of selected input and output species, it may be advantageous in some cases to facilitate the specification of the expected behaviour. Two common techniques include transformations and usage of predefined template elements.

Transformations

Transformations describe mappings between functions of different domains. As an example the *Fast Fourier Transformation* (FFT) facilitates the specification of oscillating signal behaviour by mapping domains from discrete time points to frequency. The intensity of a signal dependent on underlying oscillation frequencies is obtained. There is a variety of algorithms that calculate FFT for a given list of N signal values x_0, \dots, x_{N-1} at corresponding time points $0, \dots, N-1$. The mapping into each value X_k of the frequency domain results from

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{2\pi i}{N}nk} \quad k = 0, \dots, N-1.$$

FFT can be interpreted as a *folding* resp. *filtering* process. Other filterings useful for specification of signal processing gradients are *Discrete Wavelet Transformation* (DWT) and *Gabor Filter*. DWT is used for signal coding, where the properties of the transformation are exploited to represent a discrete signal in a more redundant form, often as a preconditioning for data compression. The DWT of a signal x is calculated by passing through a series of filters. Firstly, the samples are passed through a low pass filter with impulse response g resulting in a convolution. Simultaneously, the signal is also decomposed using a high pass filter h . Since half of the frequencies of signal x have been removed, half of the samples can be discarded. Filter outputs are then downsampled by 2.

$$y_{\text{low}}(n) = \sum_{k=-\infty}^{\infty} x_k \cdot g(2 \cdot n - k)$$
$$y_{\text{high}}(n) = \sum_{k=-\infty}^{\infty} x_k \cdot h(2 \cdot n - k)$$

Gabor filters are directly related to wavelets since they can be designed for a number of dilations and rotations. Filters can be combined forming a series or a cascade.

Usage of Template Elements

Predefined parts (subunits) of a CSN can be handled as templates. They are excluded from any kind of modifications during the evolutionary process. The advantage of this approach lies in its ability to keep stable functional units. Additional components of the CSN arising during the evolutionary process have to adopt interfaces and properties of templates. Using SBML (Systems Biology Markup Language) as a computer language for phenotypic representation of CSNs, template blocks can be declared. The experimental CSN evolver software provided by the Friedrich Schiller University within ESIGNET workpackages WP4 and WP5 comes with this feature.

7 Aspects of CSN Fitness Measure and Evaluation

Distance between Target Function and CSN Output

We assume that there exists a numeric representation of target function values $f(t_i)$ and CSN output function values $g(t_i)$ at discrete time points t_1, \dots, t_m . Let $I = \text{dom}(f) = \text{dom}(g)$. The distance $d : I \times I \rightarrow \mathbf{R}$ is called a dissimilarity measure or distance function if $\forall i, j, h \in I$:

- | | | |
|------|----------------------------------|---|
| (D1) | $d(i, j) \geq 0$ | Distances are nonnegative. |
| (D2) | $d(i, i) = 0$ | Distance of an object to itself is 0. |
| (D3) | $d(i, j) = d(j, i)$ | Distance functions are symmetric. |
| (D4) | $d(i, j) \leq d(i, h) + d(h, j)$ | Distance between i and j is not longer than a detour over h . |

Euclidian distance $\sqrt{\sum_{i=1}^m (f(t_i) - g(t_i))^2}$, its square, the *Block* distance $\sum_{i=1}^m |f(t_i) - g(t_i)|$, and the *Chebychev* distance $\max_{i=1}^m |f(t_i) - g(t_i)|$ belong to metrics often used.

Similarity between CSNs

Let $I = \{a_1, \dots, a_n\}$ be a set of objects. $s : I \times I \rightarrow \mathbf{R}$ is called a similarity measure or similarity function if $\forall i, j, h \in I$:

- | | | |
|------|-------------------------|---|
| (S1) | $0 \leq s(i, j) \leq 1$ | Similarities are between 0 and 1. |
| (S2) | $s(i, i) = 1$ | Similarity of an object to itself is 1. |
| (S3) | $s(i, j) = s(j, i)$ | Similarity functions are symmetric. |

The *Dice* coefficient $s(x, y) = \frac{2xy}{x^2 + y^2} = \frac{2\sum_{i=1}^m f(t_i)g(t_i)}{\sum_{i=1}^m f^2(t_i) + \sum_{i=1}^m g^2(t_i)}$ and the *Jaccard* coefficient $s(x, y) = \frac{xy}{x^2 + y^2 - xy} = \frac{\sum_{i=1}^m f(t_i)g(t_i)}{\sum_{i=1}^m f^2(t_i) + \sum_{i=1}^m g^2(t_i) - \sum_{i=1}^m f(t_i)g(t_i)}$ belong to metrics often used.

Distance and similarity metrics are suitable to evaluate CSN target and output functions. These techniques are also applicable to numeric representations of CSNs. It can result from *genotypic* representations of the CSNs whose nucleotides are interpreted as digits. All isomorphic representations (permutations of nodes) have to be considered and subsequently, their overall similarity is defined by the maximum similarity. Using a genotypic representation, *alignments* and *scorings* can also express homologies between CSNs.

Other approaches reflect Hamming distances, clustering, and threading. The *Hamming* distance between two CSNs can be seen as equal to the minimum number of modification steps that transform one CSN into the other. *Clustering* generates a tree of subgraphs from a CSN considering the number of edges between nodes. Nodes connected by a maximum of edges forming cluster structures are unified (melted). The minimum number of clustering steps until both CSNs have the same clustered representation also measures similarity. *Threading* is a technique coming from the comparison of protein structures. Since protein structures are denoted as bonding graphs, threading can also be adopted to CSNs.

Since CSNs as a special form of reaction networks are known to be computational complete, a direct comparison of two CSNs requires the solution of the so called equivalence problem in computer science. The equivalence problems consists in the decision whether or not either algorithmic descriptions produce the same output, considering all permitted inputs. This problem is known to be undecidable because of its infinite need of resources and the uncountable size of the search space.

Weighting of Components for Fitness Evaluation

To obtain a fitness evaluation function, all relevant components resulting from CSN properties and from the distance or similarity of its behaviour to the target function are weighted by factors and combined linearly. Further studies will explore specific weightings for CSNs.

8 Conclusions

The report summarises a general framework of CSN properties useful to specify target functions and fitness evaluation. Adopting aspects of this general framework, selected techniques for formal representation of target functions are introduced including some methods to facilitate specification. Aspects of CSN fitness measure and evaluation including distance and similarity measures complete the report.

Each of the sketched properties, techniques, and methods focuses on specific facets of interest in modelling and analysis of CSNs. Underlying theories, methodologies, and strategies are well established in the scientific community. We abstain from giving a detailed introduction into the background issues. Although some of the approaches mentioned before are known for several decades, new applications and areas of usage arise constantly. They aim to give inspirations and orientations for upcoming parts of the ESIGNET project. For further information, we refer to primary literature mentioned as well as to publications in the context of ESIGNET and its members.

References

- [1] L. von Bartalanffy. *General System Theory*. Prentice Hall, 1976 (first edition 1968)
- [2] W. Banzhaf, T. Christaller, P. Dittrich, J. Ziegler, J.T. Kim. *Advances in Artificial Life*. LNAI 2801, Springer-Verlag Berlin, Heidelberg, 2004
- [3] J.M. Berg, L. Stryer, J.L. Tymoczko. *Biochemistry*. W.H. Freeman Publishing House, 2002
- [4] E. Börger. *Specification and Validation Methods*. Oxford University Press, 1995
- [5] C. Calude, G. Păun. *Computing with Cells and Atoms*. Taylor and Francis, 2000
- [6] H. Gould, J. Tobochnik. *An Introduction to Computer Simulation Methods*. Addison Wesley, 1987
- [7] D. Knuth. *The Art of Computer Programming*. Addison Wesley, 1998
- [8] R.J. McEliece. *The Theory of Information and Coding*. Cambridge University Press, 2004
- [9] C.H. Papadimitriou. *Computational Complexity*. Addison Wesley, 1994
- [10] R. Sedgewick. *Algorithms in C: Graph Algorithms*. Addison Wesley, 2001
- [11] R.J. Trudeau. *Introduction to Graph Theory*. Dover Publications, 1994
- [12] I.H. Witten, E. Frank. *Data Mining*. Morgan Kaufmann Publishers, 2005