# Automatic termination of parallel optimization runs of stochastic global optimization methods in consensus or stagnation cases

**Jurijs Sulins[1,2*], Martins Mednis[1,2]**

[1]*Biosystems Group, Department of Computer Systems, Latvia University of Agriculture, Liela iela 2, LV-3001, Jelgava, Latvia*
[2] *SIA TIBIT, Rīgas iela 52, LV-3018, Ozolnieki, Ozolnieku nov., Latvia*
**Corresponding author*
*jurijs.sulins@hotmail.com*

Received: 8 May 2012; accepted: 19 May 2012; published online: 21 May 2012.
This paper has no supplementary material.

**Abstract:** *Dynamic models give detailed information about the influence of many parameters on the behaviour of the biochemical process of interest. Parameter optimization of dynamic models is used in parameter estimation tasks and in design tasks. A drawback of the popular family of global stochastic optimization methods is the stochastic nature of the convergence of the best value of objective function to the global optimum or a value close to that. Therefore the optimization can take long time until a stable value of objective function is reached. Even then the risk of stagnation far from global optimum remains. That sets force to look for efficient approaches to reduce optimization time and discover cases of poor performance of optimization methods.*
*Parallel optimization runs of identical optimization tasks can be used to reduce the impact of stochastic processes used in stochastic optimization methods. Consensus and stagnation criteria are proposed to terminate a set of parallel optimization runs when it is assessed that no significant improvements of the best value of the objective function are expected.*
*Four automatically detectable cases of behaviour of a group of parallel optimization runs are analysed: 1) reaching of consensus criterion (consensus case), 2) stagnation of all optimization runs without reaching the consensus criterion (stagnation case), 3) stagnation at the initial value of the objective function, 4) lack of feasible solution.*
*The proposed approach can be used automating the termination of optimization process when no further progress of the best value of objective function is expected. Suitability of particular optimization method with its settings for particular optimization task can be assessed analysing the dynamics of objective function's best values of parallel runs.*

**Keywords:** optimization, parameter estimation, design task, dynamic modelling, convergence dynamics.

## 1 Introduction

The mission of systems biology and synthetic biology in metabolic engineering tasks (Mendes and Kell, 1998) is to facilitate the development of new bioprocesses by the help of *in silico* procedures thus reducing the amount of necessary biological experiments which are more costly both in terms of time and resources.

Dynamic models give detailed information about the influence of many parameters of the network like kinetic parameters of reactions and concentrations of reactants (Stelling, 2004). The most typical approach to represent biochemical networks is through a set of coupled deterministic ordinary differential equations intended to describe the network and the production and consumption rates for the individual species involved in the network (Balsa-Canto et al., 2010). The expected increase of the size of dynamic models (Jamshidi and Palsson, 2008) will facilitate their application. Serious challenge in case of optimization of dynamic model is lack of analytical optimization solutions to solve systems of nonlinear differential equations.

Therefore the numerical methods are used in optimization tasks of biochemical networks. The numerical methods can be classified as local and global optimum seeking methods (Balsa-Canto et al., 2008; Mendes and Kell, 1998). Usually the global optimization methods are used to avoid stagnation of the solution in local optima. There are two classes of global numerical optimization methods: deterministic and stochastic. The advantage of some of deterministic methods is the guaranteed reach of global optimum for the price of unknown computation time (Banga, 2008; Moles et al., 2003). Therefore, the stochastic global optimization methods are the most popular in optimization tasks of biochemical networks due to their universality and relatively fast convergence to the global optimum close value (Banga, 2008; Moles et al., 2003).

In case of single optimization run of stochastic global optimization method the termination criterion usually is a stable best value of the objective function for a relatively long time and it cannot be determined if that is a stagnation at local optima or the best value is reached. Therefore in case of stagnation of a single optimization run at local optima misleading conclusions can be done about the optimization potential of given set of adjustable parameters (Mozga and Stalidzans, 2011c).

The convergence of global stochastic optimization methods is analysed in case of parameter estimation tasks (Baker et al., 2010; Balsa-Canto et al., 2008, 2010; Mendes and Kell, 1998; Moles et al., 2003). Convergence dynamics for design

optimization (Mendes and Kell, 1998) or more generally process optimization tasks where the properties of metabolic pathways are changed with the aim of enhancing the production of some metabolite of interest (Mendes and Kell, 1998; Moles et al., 2003) is analyzed in several recent publications (Mozga and Stalidzans, 2011b, 2011c; Mozga et al., 2011). A software tool ConvAn (Kostromins et al., 2012) for analysis of convergence dynamics suitable for both parameter estimation and design tasks has been developed for statistical analysis of performance of stochastic optimization methods. The convergence speed and reliability of optimization method are critical in design problems of biochemical networks (Mozga and Stalidzans, 2011b, 2011c) where even relatively small number (5-15) of adjustable parameters of the model cause hundreds or thousands of combinations to be explored (Mozga and Stalidzans, 2011a). The combinatorial explosion of number of adjustable parameter combinations sets force to look for efficient approaches to reduce necessary optimization time.

A set of criteria is proposed to terminate a parallel optimization runs when it is assessed that no significant improvements of the best value of objective function are expected. The first criterion is the consensus of parallel optimization runs which indicate that all the parallel optimization runs have converged via different trajectories to the same solution indicating also good performance of the optimization method (Mozga and Stalidzans, 2011b, 2011c). The second criterion is a long stagnation of all optimization runs at different best values indicating poor performance of optimization method (Mozga and Stalidzans, 2011b, 2011c; Mozga et al., 2011).

Use of proposed criteria for automatic termination of optimization both for parameter estimation and design tasks reduce the length of optimization experiment by more intensive use of computational resources due to parallel optimization runs. The main advantage compared to a single optimization run is the early detection of the best value (consensus of independent optimization runs) or bad performance of optimization (stagnation of at least one optimization run).

## 2 Matherials and methods

Yeast glycolysis models from Biomodels data base (Le Novère et al., 2006) are used to examine the performance of consensus and stagnation criteria. Criteria are demonstrated in design optimization tasks where objective function has to be maximized. Software COPASI (Hoops et al., 2006) is used as optimization tool. Parallel optimization experiments using stochastic global optimization methods with COPASI 4.7 Build 34 are automatically set and performed using software CoRunner (Sulins and Stalidzans, 2012). Since stochastic optimization methods involve use of random numbers, successive optimization runs on the same model with the same algorithm converge to the best value in a different trajectory. Convergence dynamics of optimization runs is analysed using software ConvAn (Kostromins et al., 2012).

In the maximization experiments the values are normalized the way that 0% of objective function value corresponds to the objective function value of unmodified model while 100% correspond to the best value of objective function found in any of parallel runs in particular time moment. Thus the value of objective function that correspond 0% remains constant while the value of 100% increases during optimization until the best value is reached or stagnation starts.

In case of minimization experiments the best value of objective function is decreasing and the module of changes of the best value of objective function has to be taken into account calculating 100% value.

**Consensus criterion** is fulfilled when all of parallel optimization runs reach a value of objective function which lies within pre-defined consensus corridor. The consensus corridor can be expressed in per cents: 3% corridor would mean that the best values of all parallel optimization runs have to be within 97-100% corridor. Criterion was analysed optimizing yeast glycolysis model of Galazzo and Bailey (Galazzo and Bailey, 1990) for ethanol production (Rodríguez-Acosta et al., 1999). The model contains 2 compartments, 8 reactions and 9 metabolites. Objective function in all optimization runs was to maximize flux of pyruvate kinase which is proportional to the ethanol production. Concentrations of enzymes catalysing reactions ATPase, GAP, Glucose in, Hexokinase, Phosphofructokinase and Pyruvate kinase were chosen as adjustable parameters.

Evolutionary programming optimization method (Back and Schwefel, 1993; Back et al., 1997; Fogel et al., 1992) was used with following method settings: Number of Generations: 30000; Population Size: 20; Random Number Generator: 1; Seed: 0. The values of adjustable parameters were allowed to change within a wide range from -99% up to 900% from their initial values. “Steady state” subtask of optimization within COPASI was chosen to avoid solutions without steady state.

**Stagnation criterion** is fulfilled when all the parallel optimization runs do not change their best value of objective function for a pre-set stagnation delay time while the consensus is not reached. The pre-set stagnation delay time can be defined in time units or as per cents of optimization duration. Stagnation was analysed using yeast glycolysis model of Hynne and co-workers (Hynne et al., 2001). The model contains 2 compartments, 24 reactions and 25 metabolites. Objective function in all optimization runs was

$$K = \frac{Ethanol\ flow}{Glucose\ uptake} + 5 \times Ethanol\ flow$$

The sets of adjustable parameters and the optimization method were chosen on purpose to observe the stagnation behaviour (Mozga and Stalidzans, 2011b). Concentrations of enzymes catalysing five reactions (Hexokinase, Alcohol dehydrogenase, ATP consumption, Glycerol synthesis, Phosphofructokinase) were chosen as adjustable parameters. Evolutionary programming optimization method (Back and Schwefel, 1993; Back et al., 1997; Fogel et al., 1992) was used with following method settings: Number of Generations: 30000; Population Size: 20; Random Number Generator: 1; Seed: 0. The values of adjustable parameters were allowed to change within a wide range from -99% up to 1000% from their initial values. “Steady state” subtask of optimization within COPASI was chosen to avoid solutions without steady state.

Five optimization experiments were performed for each experimental setup number of reactions for each optimization method on a server running 64-bit Microsoft Windows Server 2008 Standard Service Pack 2 operating system. Server has 4x QuadCore Intel Xeon MP E7330 2400 MHz CPU and 32768 MB of RAM. Single processor per task was used as COPASI does not support optimization with parallel task distribution.

## 3 Results and discussion

Two criteria of termination of parallel optimization runs are tested for their ability to terminate the optimization when no significant increase of the best value is expected.

Generally there are four cases of behaviour of a group of parallel optimization runs: 1) reaching of consensus criteria (consensus case), 2) stagnation of all optimization runs without reaching the consensus criteria (stagnation case), 3) consensus at the initial value of objective function, 4) lack of feasible solution.

### 3.1. Consensus

Convergence to consensus best value of the objective function indicates good performance of optimization when all the parallel runs of stochastic optimization method have reached the same or very similar best value within the consensus corridor. That is a good reason to conclude that the best value found is close to the global optimum still keeping in mind that finding global optimum cannot be guaranteed by stochastic global optimization methods (Banga, 2008; Moles et al., 2003). A consensus delay time (determined in time units or per cents of duration of optimization runs) can be used optionally to avoid coincidental short-time consensus.

Illustrative consensus experiment (Fig.1) demonstrates application of consensus criterion. In this particular case there is no further improvement after fulfilling of automatic consensus criterion. On the other hand it is not guaranteed that there will not be further improvement as the behaviour of stochastic optimization methods cannot be predicted with full confidence. To increase the confidence about correctness of automatically made decision the number of parallel runs can be increased or the consensus corridor can be narrowed. Both changes will increase the probability of longer duration of optimization.

### 3.2. Stagnation

Stagnation case means that all the parallel optimization runs do not change their value for the delay time and at least one optimization run stagnate at value which is not within the pre-set consensus corridor of the best one gives indication about risk that the optimization method does not perform well for particular optimization task. There is increased risk that also the other runs stagnate at values which are far from the optimal solution. It is suggested to test another optimization method or settings of the method to improve the performance. In case if several methods perform similar way it might indicate the peculiarity of the model or particular set of adjustable parameters (Mozga and Stalidzans, 2011c).

There is a risk of false detection of stagnation if the pre-set stagnation delay time is too short. This kind of risk can be reduced by increased delay time which increases the duration of optimization as a consequence.

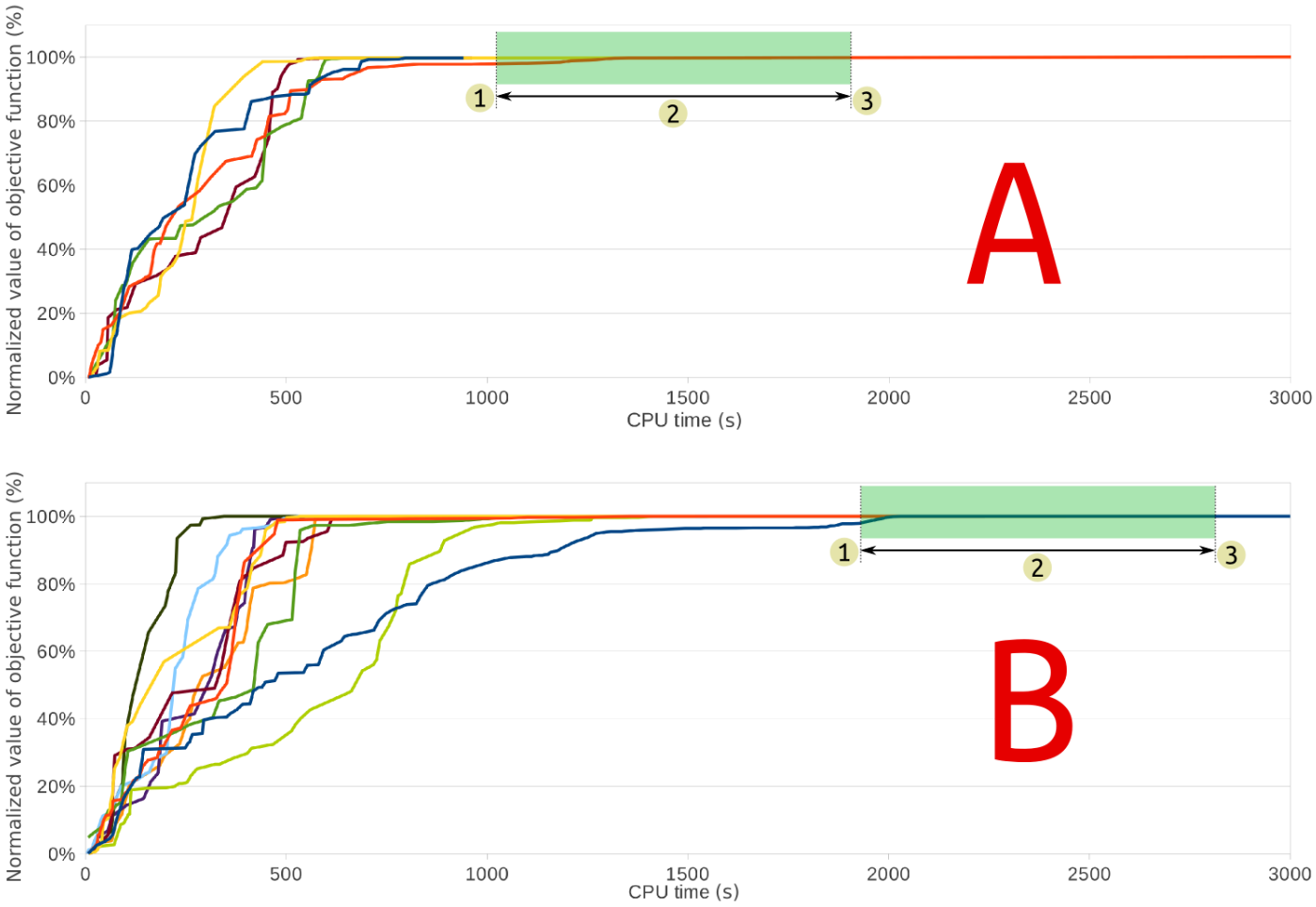


Fig. 1. **The convergence dynamics of consensus case with five (a) and ten (b) parallel optimization runs. All optimization runs have reached the 3% consensus corridor in the time moment "1". The consensus delay time "2" is 900s and lasts till the termination at the time moment "3".**

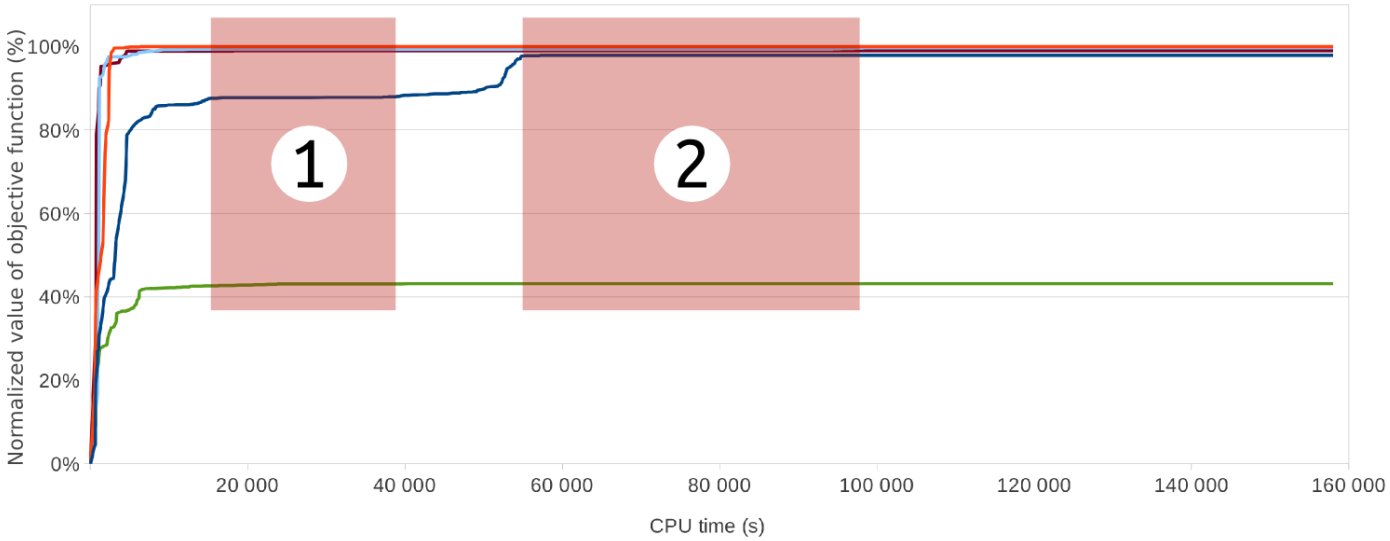


Fig. 2. **The convergence dynamics of stagnation case with five parallel optimization runs. The stagnation termination criterion is fulfilled at the end of the time period "2" which represents the stagnation delay time. In case of a shorter stagnation time (period "1") the stagnation delay time counter is reset until the next stagnation starts.**

### 3.3. Stagnation at the initial value

Stagnation of all parallel runs at the initial value of the objective function can be explained at least in two ways: 1) initial parameters of the model correspond to the best parameter values within the solution space and the optimization task is completed or 2) poor performance of optimization. The first case has very low probability. Still it cannot be fully ignored being a special case of consensus. Usually stagnation at the initial value of objective function is caused by poor performance of optimization method, huge solution space due to high number of adjustable parameters, complexity of computation because of the size or peculiarities of the model or other reasons. Improvement of optimization performance can be done by alterations of optimization method or its settings. Stagnation of all parallel runs at the initial value of the objective function is interesting as formally both consensus and stagnation criteria are reached. Therefore it is necessary to test if the value of objective function of initial model is improved to recognize this case automatically. Optimization can be terminated if initial value is not improved by any of parallel runs for some delay time.

### 3.4. Lack of feasible solution

Lack of feasible solution is a different case of stagnation at the initial value of objective function described above. Even very fast and reliably converging optimization method cannot find any solution if that is excluded by too strict or even contradicting constraints. In this case the best value usually is replaced by different expressions like "-INF", "NAN" or others in different optimization software. In this case it is useful first to check the existence of feasible steady states of the model with given constraints. If the feasible solution is not excluded by constraints, the optimization methods or their settings should be changed to improve the performance.

Stagnation criterion can detect this case automatically if the expression of objective function that corresponds to the lack of any solution with steady state in particular optimization tool is known. Automatic detection of this case should be used introducing some delay time to ensure even a small feasible area in the proposed solution space to be found.

## 4 Conclusion

Consensus and stagnation criteria of termination of parallel optimization runs of global stochastic optimization methods have been tested for their use to terminate the optimization when no significant increase of the best value of the objective function is expected. This approach can give faster and more accurate conclusion about the best value of objective function at the cost of computational resources needed for performance of parallel runs.

Consensus criterion is fulfilled when all of parallel optimization runs reach a value of objective function which lies within pre-defined consensus corridor.

Stagnation criterion is fulfilled when all the parallel optimization runs do not change their best value of objective function for a pre-set stagnation delay time while the consensus is not reached. The pre-set stagnation delay time can be defined in time units or as per cents of optimization duration.

Generally there are four automatically detectable cases of behaviour of a group of parallel optimization runs: 1) reaching of consensus criterion (consensus case), 2) stagnation of all optimization runs without reaching the consensus criterion (stagnation case), 3) stagnation at the initial value of the objective function, 4) lack of feasible solution. Optimization task can be considered as successfully completed only in the consensus case. Still also the other cases give valuable information about reasons of failure of particular setting of optimization task or optimization methods.

To reduce the risk of finding suboptimal solution the number of parallel runs can be increased or the consensus corridor can be narrowed. The side effect is the increase of probability of longer optimization duration. The probability of false detection of stagnation can be reduced by increase of the pre-set delay time causing increase of optimization duration.

## References

Back, T. and Schwefel, H.P. (1993), "An Overview of Evolutionary Algorithms for Parameter Optimization" *Evolutionary Computation*, Vol. 1 No. 1, pp. 1-23. doi:10.1162/evco.1993.1.1.1.

Back, T., Fogel, D.B., Michalewicz, Z. (1997), *Handbook of Evolutionary Computation*, Oxford: IOP Publishing/Oxford University Press, p.1130.

Baker, S.M., Schallau, K., Junker, B.H. (2010), "Comparison of different algorithms for simultaneous estimation of multiple parameters in kinetic metabolic models" *Journal of integrative bioinformatics*, Vol. 7 No. 3, pp. 1-9. doi:10.2390/biecoll-jib-2010-133

Balsa-Canto, E., Alonso, A., Banga, J.R. (2010), "An iterative identification procedure for dynamic modeling of biochemical networks" *BMC systems biology*, Vol. 4, 11. doi:10.1186/1752-0509-4-11

Balsa-Canto, E., Peifer, M., Banga, J.R., Timmer, J., Fleck, C. (2008), "Hybrid optimization method with general switching strategy for parameter estimation" *BMC systems biology*, Vol. 2, 26. doi:10.1186/1752-0509-2-26

Banga, J.R. (2008), "Optimization in computational systems biology" *BMC systems biology*, Vol. 2, 47. doi:10.1186/1752-0509-2-47

Fogel, D.B., Fogel, L.J., Atmar, J.W. (1992), "Meta-evolutionary programming" in *Proceedings of 25th Asiloma Conference on Signals, Systems and Computers*, Asilomar, pp. 540-545.

Galazzo, J.L., and Bailey, J.E. (1990), "Fermentation pathway kinetics and metabolic flux control in suspended and immobilized Saccharomyces cerevisiae" *Enzyme and Microbial Technology*, Vol. 12 No. 3, pp. 162-172. doi:10.1016/0141-0229(90)90033-M

Hoops, S., Sahle, S., Gauges, R., Lee, C., Pahle, J., Simus, N., Singhal, M., et al. (2006), "COPASI - a COmplex PAthway Simulator" *Bioinformatics (Oxford, England)*, Vol. 22 No. 24, pp. 3067-3074. doi:10.1093/bioinformatics/btl485

Hynne, F., Danø, S., Sørensen, P.G. (2001), "Full-scale model of glycolysis in *Saccharomyces cerevisiae*" *Biophysical chemistry*, Vol. 94 No. 1-2, pp. 121-63. doi:10.1016/S0301-4622(01)00229-0

Jamshidi, N., and Palsson, B.Ø. (2008), "Formulating genome-scale kinetic models in the post-genome era" *Molecular systems biology*, Vol. 4, 171. doi:10.1038/msb.2008.8

Kostromins, A., Mozga, I., Stalidzans, E. (2012), "ConvAn: a convergence analyzing tool for optimization of biochemical networks" *Biosystems*, Vol. 108 Nr. 1-3, pp. 73-77. doi:10.1016/j.biosystems.2011.12.004

Le Novère, N., Bornstein, B.J., Broicher, A., Courtot, M., Donizelli, M., Dharuri, H., Li, L., et al. (2006), "BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems" *Nucleic acids research*, Vol. *34* No. Database issue, D689-691. doi:10.1093/nar/gkj092

Mendes, P., and Kell, D.B. (1998), "Non-linear optimization of biochemical pathways: applications to metabolic engineering and parameter estimation" *Bioinformatics (Oxford, England)*, Vol.14 No.10, pp. 869-83. doi:10.1093/bioinformatics/14.10.869

Moles, C.G., Mendes, P., Banga, J.R. (2003), "Parameter estimation in biochemical pathways: a comparison of global optimization methods" in Skjoldebremd, C., Trystrom, G. (Eds.), *Genome Research*, Vol. 13 No. 11, pp. 2467-2474. doi:10.1101/gr.1262503

Mozga, I, and Stalidzans, E. (2011a), "Optimization protocol of biochemical networks for effective collaboration between industry representatives, biologists and modellers" *Proceedings of International Industrial Simulation Conference, 6-8 June 2011*, Venice, pp. 91-96.

Mozga, I, and Stalidzans, E. (2011b), "Convergence Dynamics of Biochemical Models To The Global Optimum" *Proceedings of 3rd International Conference on E-Health and Bioengineering, 24-26 November 2011, Iasi*, pp. 227-230.

Mozga, I, and Stalidzans, E. (2011c), "Convergence dynamics of biochemical pathway steady state stochastic global optimization" *Proceedings of IEEE 12th International Symposium on Computational Intelligence and Informatics (CINTI), 21-22 November 2011*, Budapest, pp. 231-235. doi:10.1109/CINTI.2011.6108504

Mozga, I, Kostromins, A., Stalidzans, E. (2011) "Forecast of Numerical Optimization Progress of Biochemical Networks" *Proceedings of International conference on Engineering for Rural Development, 26-27 May 2011*, Jelgava, pp. 103-108.

Rodríguez-Acosta, F., Regalado, C.M., Torres, N.V. (1999), "Non-linear optimization of biotechnological processes by stochastic algorithms: application to the maximization of the production rate of ethanol, glycerol and carbohydrates by Saccharomyces cerevisiae" *Journal of biotechnology*, Vol. 68 No. 1, pp. 15-28. doi: 10.1016/S0168-1656(98)00178-3

Stelling, J. (2004), "Mathematical models in microbial systems biology" *Current opinion in microbiology*, Vol. 7, No. 5, pp. 513-518. doi:10.1016/j.mib.2004.08.004

Sulins, J., and Stalidzans, E. (2012), "CoRunner: multiple optimization run manager for COPASI software" *Proceedings of International conference on Applied Information and Communication Technologies, 26-27 April 2012*, Jelgava, pp. 312 - 316.

# A Clustering Approach to Detect mRNA–Degradation Patterns from DNA–Microarray Gene-Expression Data

**Susanne Motameny[1], Röbbe Wünschiers[2*]**

[1]*Cologne Center for Genomics, Cologne University, Zülpicher Strasse 47, 50674 Cologne, Germany*
[2] *University of Applied Sciences Mittweida, Technikumplatz 17, 09648 Mittweida, Germany*
**Corresponding author*
*roebbe.wuenschiers@hs-mittweida.de*

Received: 15 October 2012; accepted: 2 November 2012; published online: 12 November 2012.
This paper has no supplementary material.

**Abstract:** *DNA-microarray based gene-expression analysis is based on hybridization events between messenger RNA (mRNA) and single stranded DNA probes. In oligo nucleotide DNA-microarrays the probes consist of approximately 20 to 80 nucleotides long DNA-molecules. Consequently, several unique probes perfectly matching each single open reading frame (ORF) of mono- or polycistronic mRNA are usually used. If these probes are distributed over the whole length of the mRNA molecule, information about mRNA-degradation patterns can be gathered with data clustering methods.*
*Here we report analysis of expression of 1107 open reading frames from the cyanobacterium* Nostoc *PCC 7120. Each open reading frame was covered by 10 unique 25 nucleotides long probes and analyzed by 4 independent DNA-microarray experiments. Both the positional information and the absolute expression value for each probe were used to infer clusters of transcripts that show similar expression patterns. Hierarchical and fuzzy k-means clustering yielded comparable results. Our results suggest that several different mRNA-degradation mechanisms, specific for certain transcripts, work in concert.*

**Keywords:** Bioinformatics, Computational Biology, DNA-Microarrays, Gene-Expression, mRNA-Degradation.

## 1 Introduction

In living organisms, information generally flows from DNA via mRNA to protein (Fig. 1).

Each protein is encoded by a gene, the expression of which is regulated by an upstream promoter region. The expression process is mediated by two steps: transcription of one (monocistronic transcript) or many (polycistronic transcript) genes to mRNA and translation of the mRNA to protein(s), respectively. As long as a mRNA molecule is present in the bacterial cell it will be translated to protein. In order to respond quickly to changing biotic or abiotic conditions, the expression process is regulated at different levels. Of major importance is the regulation of the amount of transcripts (mRNA-molecules) (Lackner and Bähler, 2008). Therefore, the transcription of genes is switched on and off or regulated up or down. These regulatory events can only take effect, if the corresponding mRNA is inactivated quickly.

One known and obvious process of transcript inactivation is mRNA-degradation, which has been analyzed intensively in the past (see, e.g. Carpousis et al., 1999; Garneau et al., 2007; Houseley and Tollervey, 2009; Kristoffersen et al., 2012). In *Escherichia coli*, mRNA-degradation is mediated by the combined action of endo- and exoribonucleases (Nierlich and Murakawa, 1996).

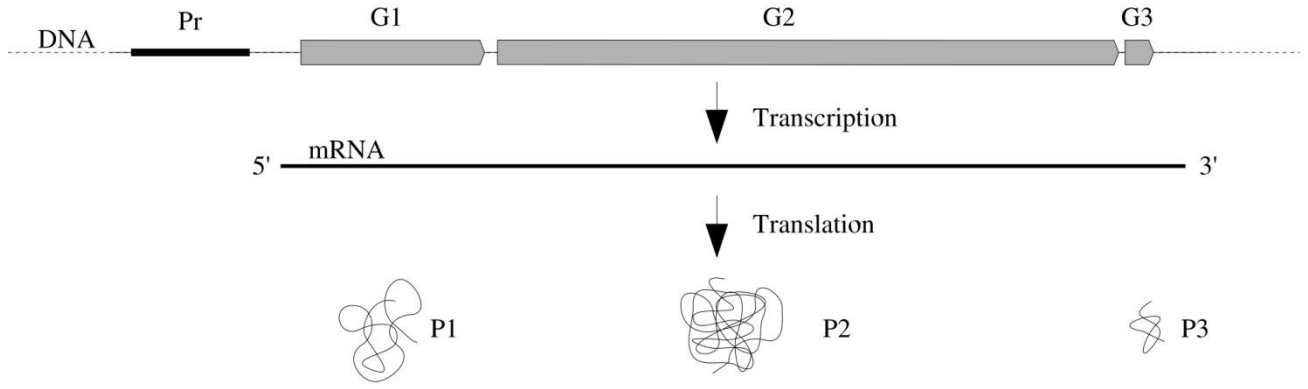


Fig. 1. **Gene Expression – The linear sequence of four different nucleotides (A, C, T, G) on the DNA carries information. Defined stretches of DNA, genes (G1, G2, G3), encode for proteins (P1, P2, P3). In bacteria, several genes are usually organized as operons and regulated by a common promoter (Pr).**

More than 20 ribonucleases have been identified in this organism. Degradation initiates with an endoribonucleolytic cleavage followed by exoribonuclease digestion to generate 5'-mononucleotides. The exoribonucleases PNPase and RNase II are key players for the 3' to 5' processive degradation, while the exoribonucleases RNase R is particularly important for removing mRNA fragments with extensive secondary structures (Cheng and Deutscher, 2005). The 5'-end dependent endonuclease RNase E catalyzes a 5' to 3' processive degradation (Mackie, 1998). Only RNase E and RNase P appear to be essential for growth since no knock-out mutant could be isolated (Donovan and Kushner, 1986). All components are organized as a large multiprotein complex, known as the RNA-degradosome. Genes encoding enzymes related to PNPase, RNase II, and RNase R can be found in the cyanobacterium *Nostoc* PCC 7120 as well.

Although transcript stability has been analyzed for some prokaryotes (Selinger et al., 2003) the variety of the corresponding mRNA-degradation pathways remain only partially characterized (Kaberdin et al., 2011). This initiated us to examine DNA-microarray based gene-expression data generated in our lab for mRNA-degradation patterns. In contrary to other DNA-microarray based studies (e.g. Selinger et al., 2003) we do not repress gene expression by application of transcriptional inhibitors but use data from one single time-point. This has the advantage of an undisturbed data set at the cost of less resolution. Each open reading frame was covered by 10 unique probes and analyzed by 4 independent DNA-microarray experiments. Both the positional information and the absolute expression value for each probe were used to infer clusters of transcripts that show similar expression patterns. Our results suggest that several different mRNA-degradation mechanisms, specific for certain transcripts, work in concert.

Based on mRNA degradation in *E. Coli*, three degradation patterns can be expected to appear in the expression data. These are illustrated in Fig. 2.

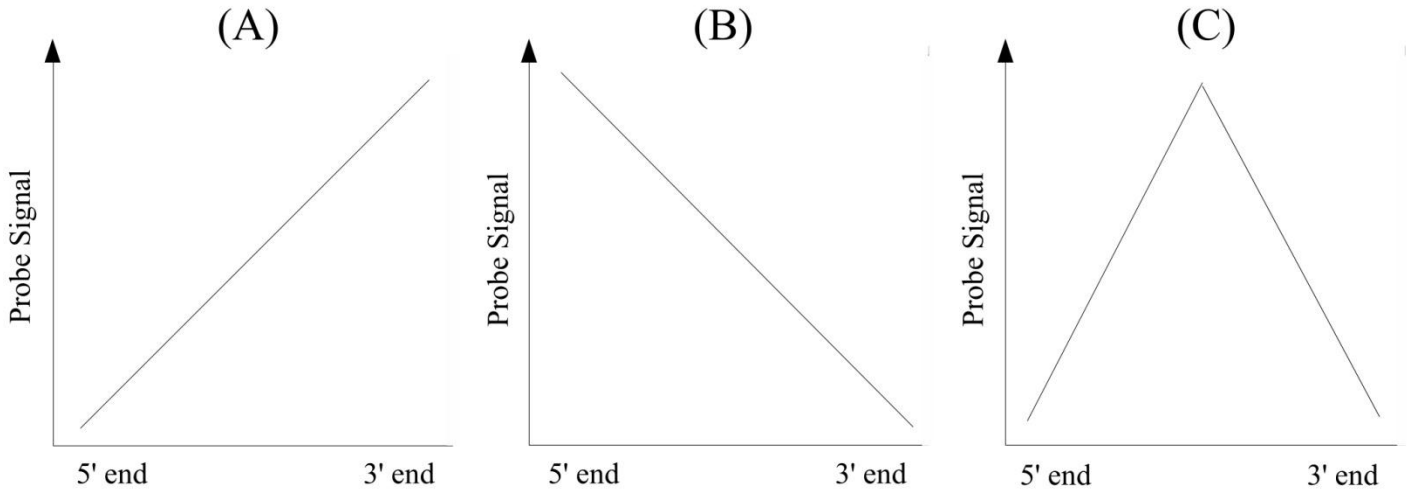


Fig. 2. **Expected Degradation Patterns – (A) 5' to 3' degradation, (B) 3' to 5' degradation, (C) simultaneous degradation from 5' to 3' and 3' to 5'**

## 2 Matherials and methods

### 2.1. Strain and Culture Conditions

The cyanobacterium *Nostoc* sp. strain PCC 7120 (*Nostoc* PCC 7120; formerly *Anabaena* PCC 7120) was grown in either nitrogen fixing or non-nitrogen fixing conditions as previously described (Hansel et al., 2001).

### 2.2. Preparation of Biotin Labeled, Fragmented cRNA

Total RNA from *Nostoc* PCC 7120 was extracted as previously described (Axelsson and Lindblad, 2002). From 10 µg of total RNA, low molecular weight RNA, e.g. tRNA and 5S rRNA, was removed by size exclusion chromatography (MEGAclear kit, Ambion). To remove 16S and 23S rRNA, the MICROBExpress kit from Ambion was used. The remaining RNA was linearly amplified by a modified Eberwine protocol (Eberwine et al., 1992) as follows. If not differently stated, all enzymes and chemicals were purchased from Invitrogene.

**First Strand Synthesis**. The pelleted RNA from the previous mRNA enrichment steps was resuspended in 4.25 µl water and mixed with 1 µl of T7 random hexamers (0.5 µg/µl; 5'-GGC CAG TGA ATT GTA ATA CGA CTC ACT ATA GGG AGG CGG NNN NNN-3'). Following incubation at 70ºC for 10 min, 4ºC for 2 min and 23ºC for 5 min, 3.75 µl reaction mix (2 µl 5x first strand synthesis buffer, 1 µl 0.1 M DTT, 0.5 µl 10 mM dNTP mix, 0.25 µl 40 U RNase OUT and 200 U Superscript II polymerase) was added to the RNA/primer mix. First strand synthesis reaction was performed with the following temperature scheme: 37ºC for 20 min, 42ºC for 20 min, 50ºC for 15 min, 55ºC for 10 min and 65ºC for 15 min. After adding 0.5 µl RNase H, the reaction mix was incubated for another 30 min at 37ºC and 2 min at 95ºC. Then, 1.7 µl random hexamer primers (0.3 µg/µl) were added and the mix incubated for 10 min at 70ºC.

**Second Strand Synthesis.** The product of the first strand synthesis was mixed with 43.8 µl water, 15 µl 5x second strand synthesis buffer, 20 U DNA polymerase I, 1.5 µl 10 mM dNTP and 1 U RNaseH and incubated for 2 h at 16ºC. After addition of 10 U T4 DNA-polymerase the reaction mix was first incubated at 16ºC for 15 min and then at 70ºC for 10 min.

**Isolation of ds-cDNA.** Double stranded cDNA was isolated from the product of second strand synthesis according to standard procedures (Maniatis et al., 1982).

***In vitro* Transcription.** The pelleted ds-cDNA was resuspended in 1.5 µl water. The MEGAscript T7 kit (Ambion) was used for *in vitro* transcription. In addition to the standard nucleotides, 3.75 µl 10 mM Bio-16-CTP (NEN) and 3.75 µl 75

mM Bio-11-UTP (Roche) were added to the reaction mix. This led to the formation of biotinylated cRNA.

**cRNA Isolation.** The RNeasy kit (Qiagen) was applied for cRNA isolation. All steps were performed according to the manufacturer's instructions.

**cRNA Fragmentation.** For cRNA fragmentation 15 µg cRNA was resuspended in 2.5 µl water and 2.5 µl 2x fragmentation buffer (5x stock: 200 mM Tris, 150 mM Mg-acetate, 500 mM K-acetate, pH 8.1). The reaction mix was incubated for 5 min at 94ºC. The fragmentation reaction was performed immediately prior to hybridization.

### 2.3. Oligonucleotide Probe Selection

A unique *Nostoc* PCC 7120 probe set (as many 25mer probes per open reading frame (ORF) as possible) was calculated based on the full genome sequence (retrieved online from CyanoBase: http://www.kazusa.or.jp/cyanobase/Anabaena/index.html) using a combination of sequence uniqueness criteria and rules for selection of oligonucleotides likely to hybridize with high specificity and sensitivity. The selection criteria were essentially as described in Lockhart *et al.* (Lockhart et al., 1996) with modifications for the longer probes used here (25mers instead of 20mers). If available, 10 unique probes per ORF were used in the experiments.

### 2.4. DNA-Microarray Production and *In Situ* Oligo Nucleotide Synthesis

Light-activated *in situ* oligonucleotide synthesis was performed essentially as described by Singh-Gasson *et al.* (Singh-Gasson et al., 1999) using a digital micromirror device (Güimil et al., 2003). The synthesis was performed within the geniom one device (Febit AG, Heidelberg, Germany) on an activated three-dimensional reaction carrier consisting of a glass-silica-glass sandwich (DNA processor). Four individually accessible microchannels (referred to as arrays) etched into the silica layer of the DNA processor are connected to the microfluidic system of the geniom device. Using standard DNA synthesis reagents and 3Â´-phosphoramidites with a photolabile protecting group (Beier and Hoheisel, 2000; Hasan et al., 1997), oligonucleotides were synthesized in parallel in all four translucent arrays of one reaction carrier. Prior to synthesis, the glass surface was activated by coating with a silane-bound spacer. The probe sets synthesized within the four arrays may be the same but also can be different on all arrays.

### 2.5. Hybridization

Hybridization was performed with 7.5 µg fragmented cRNA (see above) in a final volume of 10 µl. Hybridization solutions contained 100 mM MES (pH 6.6), 0.9 M NaCl, 20 mM EDTA and 0.01% (v/v) Tween 20. In addition, the solutions contained 0.1 mg/ml sonicated herring sperm DNA and 0.5 mg/ml BSA. RNA samples were heated in the hybridization solution to 95ºC for 3 min followed by 45ºC for 3 min before being placed in an array which had been prehybridized for 15 min with 1% (w/v) BSA in hybridization solution at RT. Hybridizations were carried out at 45ºC for 16 h. After removing the hybridization solutions, arrays were first washed with non-stringent buffer (0.005% (v/v) Triton X-100 in 6 x SSPE) for 20 min at 25ºC and subsequently with stringent buffer (0.005% (v/v) Triton X-100 in 0.5 x SSPE) for 20 min at 45ºC. After washing, the hybridized RNA was fluorescence-stained by incubating with 10 µg/ml streptavidin-phycoerythrin and 2 µg/µl BSA in 6 x SSPE at 25ºC for 15 min. Unbound streptavidin-phycoerythrin was removed by washing with non-stringent buffer for 20 min at 25ºC. Hybridizations were not performed competitive on one DNA processor but separated with one condition per DNA processor.

### 2.6. Detection & Raw Data Generation

The CCD-camera based fluorescence detection system, equipped with a Cy3 filter set, integrated into the geniom one automate was used. 36 pixels per spot were available for data analysis.

Processing of raw data, including background correction, array to array normalization and determination of gene expression levels, as well as calculation of fold-change values were performed as described before (Zhou and Abagyan, 2002). All steps were carried out using the PROP algorithm of the geniom application software which is based on the MOID algorithm (Zhou and Abagyan, 2002).

Background correction is based on probes with no corresponding mRNA target and the average of the lowest 5% expressed genes. Data normalization is based on iteratively correcting the raw data on non-regulated genes (fold-changes less than ±2). In a comparative study of *Saccharomyces cerevisiae* gene expression with three independent techniques (i.e., Affymetrix GeneChips, geniom one microarrays, and cDNA microarrays) it was previously shown that expression fold changes with values greater than ±1.5 are significant for the geniom one technology applied here (Baum et al., 2003). Thus, genes with fold changes between -1.5 and 1.5 can be considered to be non-regulated. In our study we extend the rule such that the upper or lower bound must be greater than ±2 for upregulated and downregulated genes, respectively. Furthermore, this rule must be fulfilled for all independent experiments.

### 2.7. Data Filtering

In order to investigate mRNA-degradation it is necessary that the probes are positioned along the whole length of the transcript. Therefore, only such genes were chosen which have a probe situated within the first and last 50 bases of their transcript. 199 genes of the 1107 measured probe sets satisfy this condition and were selected for subsequent analysis. 23 of these genes showed different probe rankings on different arrays and were therefore excluded from further analysis. There are many more effects other than mRNA-degradation that influence the measured probe-transcript hybridization strength, which are not yet fully understood (Rule et al., 2009). However, it is known that the GC-content of a probe influences the strength of the hybridization and thus the expression value. To account for this effect, all genes were removed whose expression profiles showed a high Pearson's correlation (greater than 0.4) to the GC-contents of the probes. In this step, another 43 genes were excluded from further analysis. The expression values of the remaining 133 genes were logarithmized and then scaled to have mean zero and unit variance. The last step of data filtering included the removal of outliers. Outliers were defined as genes whose scaled expression profiles were dominated by a peak at a single probe. 46 genes with a Pearson's correlation higher than 0.7 with a single-peaked profile were removed.

After the filtering procedure, scaled probe profiles of 87 genes remained and were used for further analysis.

### 2.8 Clustering

Two clustering methods were applied to the filtered data: a hierarchical clustering using the GeneCluster and TreeView software from M. Eisen (Eisen et al., 1998) and fuzzy k-means (see e.g. Kruse et al., 1999), which we implemented in MATLAB. The average linkage method was chosen for hierarchical clustering, using the uncentered Pearson correlation as a similarity measure. Fuzzy k-means has been reported to be a useful method to uncover clusters in gene expression data (Gasch and Eisen, 2002). The fuzzy approach has several advantages over common hard clustering methods. It is able to detect overlapping clusters by assigning membership degrees to the genes. This means that every gene can belong to several clusters at different degrees. Another advantage is that by setting a membership cutoff, one can extract the genes which are most typical for a cluster. For the fuzzy k-means method we chose the number of clusters to be 18 and a fuzzyfier value of 1.2. The distance measure used was the euclidean distance.

## 3 Results and discussion

### 3.1. Probe Set Ranking

In order to test whether our data set is stable enough to provide an insight into mRNA-degradation, we tested if the ranking of the probes for each ORF was the same for all four independent hybridizations. Therefore, all ORF-specific probes were numbered according to their position from the 5'- to the 3'-end. Then, the probe numbers were ordered by the expression level (Fig. 3). The resulting order was compared for all four independent hybridizations. 994 of all 1107 analyzed ORFs passed this test. Out of these, another 907 expression datasets were removed as described in 2.7.

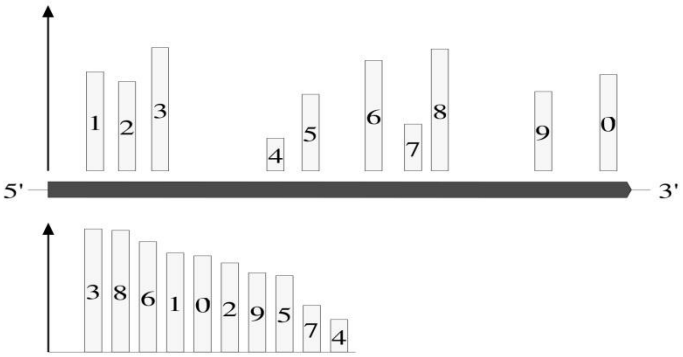


Fig. 3. **Probe Set Ranking – Schematic outline for the method used to rank probe sets for each gene for all four DNA-microarrays by their expression data. 89.8% probe sets showed the same ranking in all four individual hybridization reactions.**

### 3.2. Probe Position Dependent Expression Data

As stated above, mRNA-degradation is mediated by the combined action of endo- and exoribonucleases. Up-to-now, mRNA-degradation from the 3'- to 5'-end or *vice versa* can be discriminated. Other processes like degradation from both ends simultaneously or any other patterns have not been described yet. Under the assumption that the majority of all transcripts are degraded by the same mechanism, all ORF-specific probe sets should yield similar expression patterns. This was analyzed by testing for all ORFs if the expression value of probe *n*-1 is smaller or larger than the expression value of probe *n*. The results are shown in a matrix in Fig. 4. The number of cases where an upstream probe shows a lower expression value are counted in the upper triangle and *vice versa*.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | - | 557 | 559 | 589 | 584 | 572 | 588 | 544 | 556 | 591 |
| 2 | 550 | - | 549 | 588 | 594 | 577 | 573 | 562 | 561 | 561 |
| 3 | 548 | 558 | - | 594 | 589 | 585 | 586 | 548 | 564 | 578 |
| 4 | 518 | 518 | 513 | - | 559 | 537 | 545 | 533 | 516 | 545 |
| 5 | 523 | 513 | 518 | 548 | - | 537 | 565 | 528 | 526 | 546 |
| 6 | 535 | 530 | 522 | 570 | 570 | - | 551 | 552 | 523 | 563 |
| 7 | 519 | 534 | 521 | 562 | 542 | 556 | - | 544 | 529 | 524 |
| 8 | 563 | 545 | 559 | 574 | 579 | 555 | 563 | - | 550 | 546 |
| 9 | 551 | 546 | 543 | 590 | 581 | 584 | 578 | 557 | - | 570 |
| 0 | 516 | 546 | 529 | 562 | 561 | 544 | 582 | 561 | 537 | - |

Fig. 4. **Probe Position Dependent Expression Data – For all genes with 10 unique probes the probes were ordered from the 5'- to 3'-end (probes 1 to 0). The number of cases where an upstream probe shows a lower (upper triangle) or higher (lower triangle) expression value than the corresponding downstream probe are shown.**

If there was a common pattern from either side for the majority of the ORFs, one should observe a clear difference in the upper and the lower triangles. This is not the case, indicating that mRNA-degradation underlies no common mechanism in *Nostoc* PCC 7120. Fig. 5 shows a graphical representation of the data from Fig. 4. Here it becomes clear that mRNA-degradation underlies no common mechanism in *Nostoc* PCC 7120.

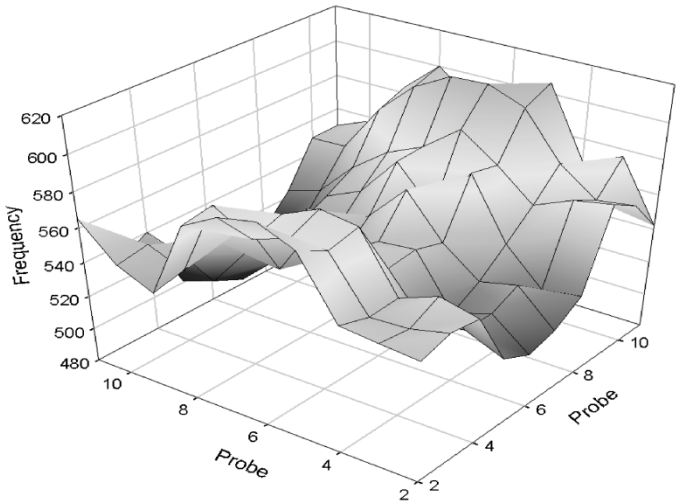


Fig.5. **Probe Position Dependent Expression Data – Graphical presentation of the data shown in Fig. 4.**

### 3.3. Clustering

#### 3.3.1. Clusters with Identical Probe Patterns

Initially, we extracted all probe sets that showed the same probe distribution patterns. We found four such clusters. 18 members of these clusters are annotated tranposases whereas 2 members are hypothetical proteins. Excluding the latter, all clusters can be explained by ORF sequence and consequently probe sequence similarity (Fig. 6).

To our surprise, these transposases are expressed at a rather high level, with equal expression levels and profiles within each cluster (Fig. 7). Transposons, also known as jumping genes, can spread themselves in a genome by a kind of copy-paste mechanism, catalyzed by a transposase enzyme for which they encode. If transposons copy themselves into a functional ORF or regulatory region, this gene commonly gets inactivated.

With our method we can discriminate four different transposon classes. Interestingly, each cluster is not only specified by a common probe pattern and similar sequences but also by similar expression values.

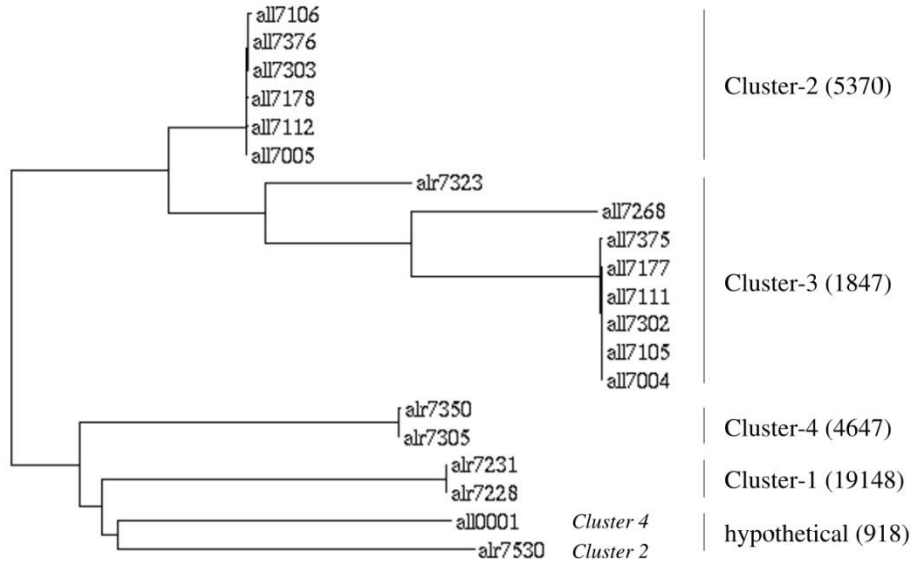


Fig. 6. **Sequence Distance Tree – This tree represents a protein sequence based distance tree of all sequences that are members of clusters with identical probe patterns. For the two hypothetical protein sequences the cluster membership is indicated. Numbers in parenthesis give the mean expression value of the members of the cluster.**

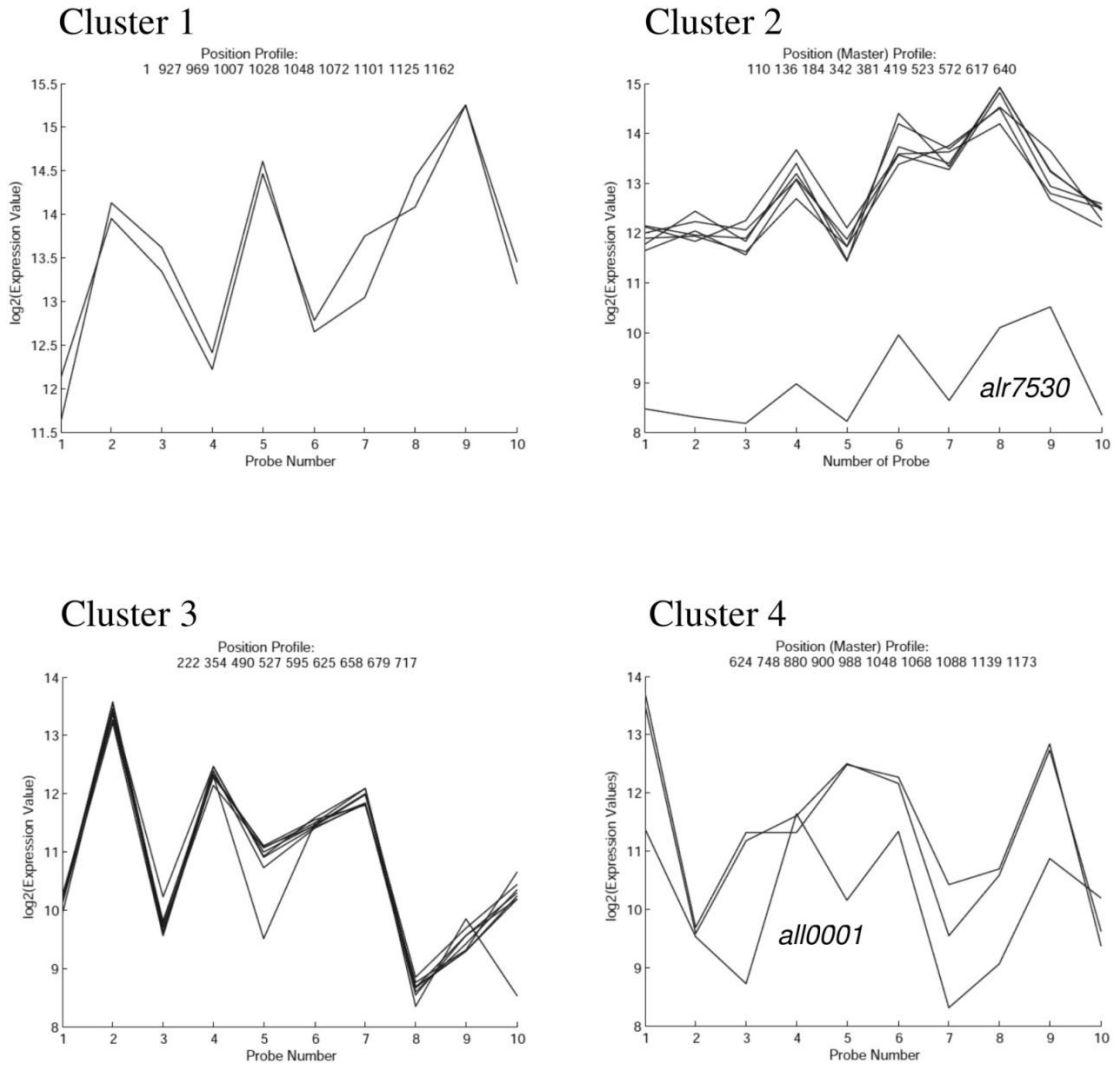


Fig. 7. **Gene Expression – Clusters with very similar probe patterns. Outliers are identified by their gene ID.**

### 3.3.2. Fuzzy K-Means

The clusters produced by the fuzzy k-means clustering are depicted in (Fig.8). Together with the cluster identifier, the number of genes within each cluster is given. The membership cutoff was set to 70%.

Clusters C5, C9, C14, and C16 show an increase of probe expression values from the 5'- to the 3'-end of the transcript. This corresponds to the assumption that a degradation mechanism is present which progresses from the 5'- to the 3'-end. Cluster C17 shows an opposite pattern, with probe expression values decreasing towards the 3'-end. This suggests a second mechanism of degradation working from the 3'-end towards the 5'-end of the transcript. A detailed view of these potentially mRNA-degradation associated clusters is shown in Fig. 9.

There are other clusters present, which show patterns that do not indicate a specific direction of degradation. It is therefore possible that degradation mechanisms other than the directional processes play a role in *Nostoc* PCC 7120 as well. However, these patterns are not very prominent in the data. There are two possible reasons for this. Firstly, genes that pass the filtering steps have relatively short transcripts. If degradation is a fast process, most transcripts will already be in a pretty degraded state and directional patterns will not be visible in the data. Secondly, effects other than degradation obscure the degradation signal in the data. Genes were filtered to have little correlation with the GC-contents of their probes, but other effects influencing the hybridization strength of the probe-target complex and their contribution to the expression patterns remain to be identified.

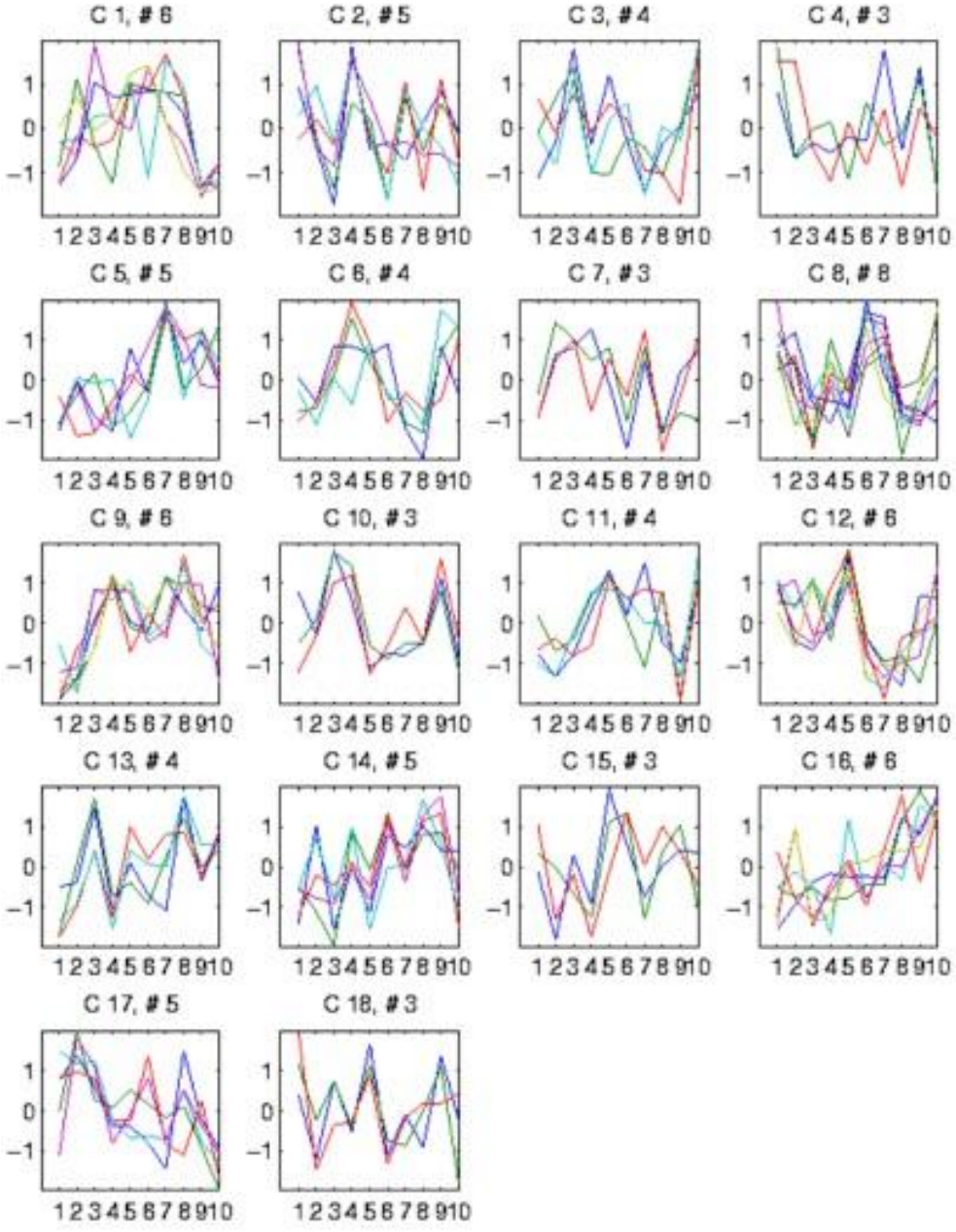


Fig. 8. **Resulting Clusters of Fuzzy K-Means Clustering – Clusters C5, C9, C14, and C16 hint at a degradation mechanism progressing from the 5'- to the 3'-end of the transcript. Cluster C17 points to a degradation process in the opposite direction.**

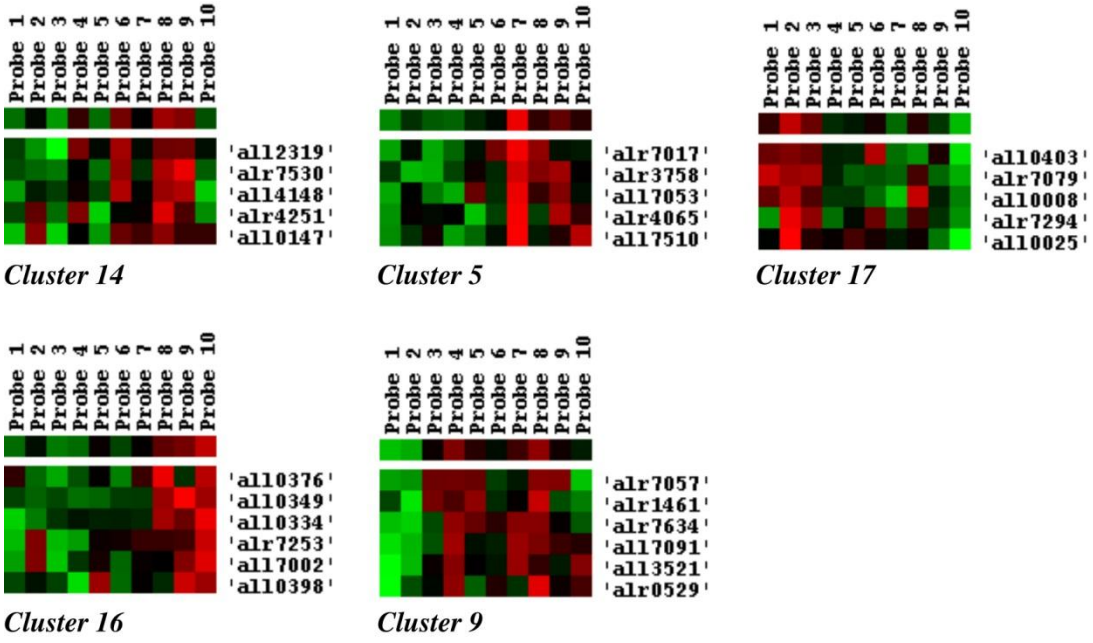


Fig. 9. **Fuzzy K-Means Clustering – Detailed view of the clusters that are potentially associated with mRNA-degradation.**

### 3.3.3. Hierarchical Clustering

The results of the hierarchical clustering are depicted in Fig. 10.

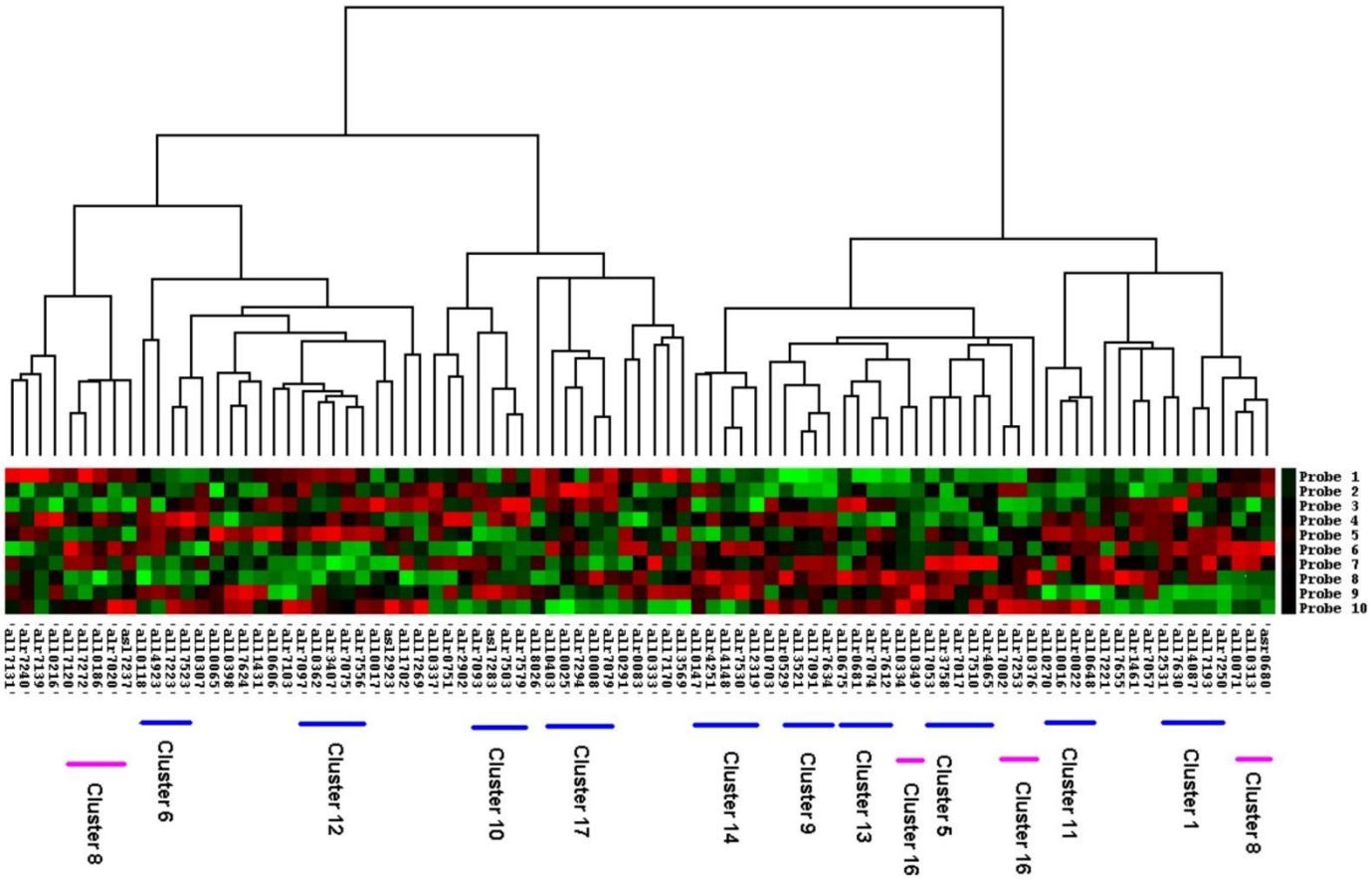


Fig. 10. **Hierarchical Clustering – Clusters also found by fuzzy k-means are marked in blue. Pink markers indicate fuzzy k-means clusters that were split by the hierarchical method.**

Clusters from the fuzzy k-means clustering are marked. The hierarchical clustering reveals no dominant structures in the data. However, it shows some overlap with the clusters produced by fuzzy k-means. Some of the fuzzy clusters are also found by the hierarchical clustering, some are separated and some are not found. Generally speaking, the hierarchical clustering does not reveal much structure, which is not unexpected because all patterns in the data are rather subtle and probably overlayed by noise.

## 4 Conclusion

The goal of the work presented here was to identify different mRNA-degradation patterns from DNA-microarray based gene-expression data. We expected to see at least 4 different degradation patterns: from 5'→3', 3'→5' and simultaneously from both ends resulting from exonucleases; and random patterns resulting from the action of endonucleases. Thus, the number of expected clusters when clustering all expression data was greater than 4 (transcripts showing no degradation where excluded from data analysis). Since gene-expression data are very noisy, we increased the number of expected clusters to 18 for fuzzy k-means clustering.

To our surprise, we did not see a big difference between hierarchical and fuzzy k-means clustering (see Fig.10). Thus, we restrict ourselves to hierarchical clustering because it does not require specification of the number of expected clusters.

Detailed analysis of our clustering results suggest that several different mRNA-degradation pathways work in concert. Furthermore, we conclude that at least some of the pathways are transcript specific. This requires the recognition of the transcript by components of the degradation machinery. Consequently, we initiated a search for common sequence and structure patterns in each individual cluster.

## Acknowledgments

We like to thank Rikard Axelsson for cell growth and RNA-purification, Michael Baum and Nicole Rittner for performing the DNA-microarray experiments, Long Li for data processing, and Ralf Müller and Alexander Schönhuth for valuable discussion. Furthermore, we like to thank two anonymous reviewers for helpful comment that helped improving the manuscript.

## References

Axelsson, R. and Lindblad, P. (2002), "Transcriptional regulation of *Nostoc* hydrogenases: effects of oxygen, hydrogen, and nickel" *Appl Environ Microbiol*, 68(1):444–7.

Baum, M., Bielau, S., Rittner, N., Schmid, K., Eggelbusch, K., Dahms, M., Schlauersbach, A., Tahedl, H., Beier, M., Güimil, R., Scheffler, M., Hermann, C., Funk, J.-M., Wixmerten, A., Rebscher, H., Hönig, M., Andreae, C., Büchner, D., Moschel, E., Glathe, A., Jäger, E., Thom, M., Greil, A., Bestvater, F., Obermeier, F., Burgmaier, J., Thome, K., Weichert, S., Hein, S., Binnewies, T., Foitzik, V., Müller, M., Stähler, C. F., and Stähler, P. F. (2003), "Validation of a novel, fully integrated and flexible microarray benchtop facility for gene expression profiling" *Nucleic Acids Res*, 31(23):e151.

Beier, M. and Hoheisel, J. D. (2000), "Production by quantitative photolithographic synthesis of individually quality checked DNA microarrays" *Nucleic Acids Res*, 28(4):E11.

Carpousis, A., Vanzo, N., and Raynal, L. (1999), "mRNA degradation. A tale of poly(A) and multiprotein machines" *Trends Genet*, 15(1):24–8.

Cheng, Z.-F. and Deutscher, M. (2005), "An important role for RNase R in mRNA decay" *Mol Cell*, 17(2):313–8.

Donovan, W. and Kushner, S. (1986), "Polynucleotide phosphorylase and ribonuclease II are required for cell viability and mRNA turnover in *Escherichia coli* K-12" *Proc Natl Acad Sci USA*, 83(1):120–4.

Eberwine, J., Yeh, H., Miyashiro, K., Cao, Y., Nair, S., Finnell, R., Zettel, M., and Coleman, P. (1992), "Analysis of gene expression in single live neurons" *Proc Natl Acad Sci USA*, 89(7):3010–4.

Eisen, M., Spellman, P., Brown, P., and Botstein, D. (1998), "Cluster analysis and display of genome-wide expression patterns" *Proc Natl Acad Sci USA*, 95:14863–14868.

Garneau, N., Wilusz, J., and Wilusz, C. (2007), "The highways and byways of mRNA decay" *Nature reviews Molecular cell biology*, 8(2):113–126.

Gasch, A. and Eisen, M. (2002), "Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering" *Genome Biology*, 3(11).

Güimil, R., Beier, M., Scheffler, M., Rebscher, H., Funk, J., Wixmerten, A., Baum, M., Hermann, C., Tahedl, H., Moschel, E., Obermeier, F., Sommer, I., Büchner, D., Viehweger, R., Burgmaier, J., Stähler, C., Müller, M., and Stähler, P. (2003), "Geniom technology–the benchtop array facility" *Nucleosides Nucleotides Nucleic Acids*, 22(5-8):1721–3.

Hansel, A., Axelsson, R., Lindberg, P., Troshina, O., Wünschiers, R., and Lindblad, P. (2001), "Cloning and characterisation of a *hyp* gene cluster in the filamentous cyanobacterium *Nostoc* sp. strain PCC 73102" *FEMS Microbiol Lett*, 201(1):59–64.

Hasan, A., Stengele, K.-P., Giegrich, H., Cornwell, P., Sham, K., Sachleben, R., Pfleiderer, W., and Foote, S. (1997), "Photolabile protecting groups for nucleotides: synthesis and photodeprotection rates" *Tetrahedron*, 53:4247–4264.

Houseley, J. and Tollervey, D. (2009), "The many pathways of RNA degradation" *Cell*, 136(4):763–776.

Kaberdin, V., Singh, D., and Lin-Chao, S. (2011), "Composition and conservation of the mRNA-degrading machinery in bacteria" *Journal of biomedical science*, 18:23.

Kristoffersen, S., Haase, C., Weil, M., Passalacqua, K., Niazi, F., Hutchison, S., Desany, B., Kolstø, A.-B., Tourasse, N., Read, T., and Økstad, O. (2012), "Global mRNA decay analysis at single nucleotide resolution reveals segmental and positional degradation patterns in a Gram-positive bacterium" *Genome Biology*, 13(4):R30.

Kruse, R., Klawonn, F., and Höppner, F. (1999), *Fuzzy Cluster Analysis*. Wiley & Sons.

Lackner, D. and Bähler, J. (2008), "Translational control of gene expression from transcripts to transcriptomes" *International Review of Cell and Molecular Biology*, 271:199–251.

Lockhart, D., Dong, H., Byrne, M., Follettie, M. T., Gallo, M., Chee, M., Mittmann, M., Wang, C., Kobayashi, M., Horton, H., and Brown, E. (1996), "Expression monitoring by hybridization to high-density oligonucleotide arrays" *Nat Biotechnol*, 14(13):1675–80.

Mackie, G. (1998), "Ribonuclease E is a 5'-end-dependent endonuclease" *Nature*, 395(6703):720–3.

Maniatis, T., Fritch, E., and Sambrook, J. (1982), "Synthesis and cloning of cDNA" In *Molecular cloning*, pages 212–246. Cold Spring Harbor Laboratory.

Nierlich, D. and Murakawa, G. (1996), "The decay of bacterial messenger RNA" *Prog Nucleic Acid Res Mol Biol*, 52:153–216.

Rule, R., Pozhitkov, A., and Noble, P. (2009), "Use of hidden correlations in short oligonucleotide array data are insufficient for accurate quantification of nucleic acid targets in complex target mixtures" *Journal of microbiological methods*, 76(2):188–195.

Selinger, D., Saxena, R. M., Cheung, K., Church, G., and Rosenow, C. (2003), "Global RNA half-life analysis in *Escherichia coli* reveals positional patterns of transcript degradation" *Genome Res*, 13(2):216–23.

Singh-Gasson, S., Green, R., Yue, Y., Nelson, C., Blattner, F., Sussman, M., and Cerrina, F. (1999), "Maskless fabrication of light-directed oligonucleotide microarrays using a digital micromirror array" *Nat Biotechnol*, 17(10):974–8.

Zhou, Y. and Abagyan, R. (2002), "Match-only integral distribution (MOID) algorithm for high-density oligonucleotide array analysis" *BMC Bioinformatics*, 3(1):3.

# Application of string similarity ratio and edit distance in automatic metabolite reconciliation comparing reconstructions and models

**Martins Mednis[1*], Maike Aurich[2]**

[1]*Biosystems group,Department of Computer Systems, Latvia University of Agriculture, Liela iela 2, LV3001, Jelgava, Latvia*
[2]*Center for Systems Biology, University of Iceland, Sturlugata 8, IS101, Reykjavik, Icland*
**Corresponding author*
*Martins.mednis@llu.lv*

Received: 4 November 2012; accepted: 12 November 2012; published online: 13 November 2012.
This paper has no supplementary material.

**Abstract:** *Increasing size and number of biochemical network models facilitates iterative way of model building. In parallel comparison of models become more important to find out the similarity of models, agreement between models and other features. Comparison of models would be convenient if all the model builders would use the same formats and names to describe the network. The reality is that models can be described in different formats (SBML, COBRA and others). The formulas of metabolites are not always indicated. IDs and names of metabolites are different even for the same metabolite.*
*A model comparison algorithm for SBML and COBRA format models is developed to match the same metabolites of different models which is precondition for correct matching of reactions.*
*The algorithm is based on comparison of metabolite names as text strings. Automatic three level filtering approach is implemented in the software ModeRator to reject pairs of potentially equal metabolites and build opinion about metabolite pairs which have high similarity in metabolite names. Results of automatic mapping were inspected with manual curation.*
*Automatic metabolite mapping of two E.coli models (1314 and 1704 metabolites) comparing only identifiers revealed high number of matching metabolites. Since both models are coming from the same source (BioCyc database) no significant difference between automatic mapping and manual curation was observed.*
*In case of two S.cerevisiae models (679 and 1061 metabolites) three level filtration by metabolite name is used. Manual curation of automatic comparison results revealed 7% discrepancy.*

**Keywords:** Biochemical networks, reconstruction, models, metabolite mapping, pairwise comparison.

## 1 Introduction

The function of cells is based on complex networks of interacting chemical reactions carefully organized in space and time. These biochemical reaction networks produce observable cellular functions (Palsson, 2006). Reconstruction of biochemical network is an assembly of the components and their interconversions for an organism, based on the genome annotation and the bibliome (Lewis et al., 2009). Reconstruction based models can be used for the analysis of network capabilities, prediction of cellular phenotypes and in silico hypothesis generation (Lewis et al., 2009).

The first fully sequenced genome was that of H.influenzae in 1995 (Fleischmann et al., 1995), which enabled the first reconstruction of a genome-scale metabolic network in 1999 (Edwards and Palsson, 1999). With the sequencing of complete genomes, it is now possible to reconstruct the network of biochemical reactions in many organisms. Several of these networks are available online: Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa and Goto, 2000; Kanehisa et al., 2011), EcoCyc (Keseler et al., 2011), BioCyc (Karp et al., 2005; Karp et al., 2010) and metaTIGER (Whitaker et al., 2009).

A potential strategy to obtain large cell models is to construct them “bottom-up” from smaller modules that can be fitted and understood more easily (Schulz et al., 2006). Bottom-up metabolic network reconstructions have been developed over the last 10 years. These reconstructions represent structured knowledge bases that abstract pertinent information on the biochemical transformations taking place within specific target organisms (Thiele and Palsson, 2010). The reconstruction process for metabolic networks has been developed (Schellenberger et al., 2011) and implemented for a number of different organisms (Palsson, 2006; Schellenberger et al., 2010) including human (Duarte et al., 2007). The available reconstructions and models are growing both in number and size (number of interactions within reconstruction or model). Still reconstructions differ in quality and coverage that may minimize their predictive potential and use as knowledge bases (Thiele and Palsson, 2010). The process of reconstruction is iterative. Unlike genome sequencing projects which have a well-defined end point, the reconstructions process is ongoing (Palsson, 2006). Thus it is important to compare, intersect and unit the existing reconstructions of the same or even different organisms to find out their quality, consistency and suitability for given task. Manual comparison becomes inefficient especially in case of genome-scale reconstructions with thousands of involved reactions and metabolites.

Genome-size reconstructions are available in different formats. Mostly they are in form of 1) plain text files, 2) SBML (Systems Biology Markup Language (Hucka et al.,

2003)) file format and 3) spreadsheets (including COBRA format (Schellenberger et al., 2011)).

Due to different formats of models and different approaches to standardization of substances and reactions the comparison of reconstructions and models is complicated. It is possible to compare structures of models visually (Boele et al., 2012; Kostromins and Stalidzans, 2012; Schellenberger et al., 2010) or parameters of the structure (Rubina and Stalidzans, 2010; Yamada and Bork, 2009). Still that is not enough to compare the scope of models or join the models as equal metabolites and reactions have to be recognized for that purpose.

Consequently, the need for analysis, comparison, merge, union and intersection of biomodels is growing. In systems biology the need to compare models or to couple them as parts of larger models has been noted by Radulescu (Radulescu et al., 2008). Also the demand for a method to relate different models has been pointed out by Gay (Gay et al., 2010).

Although we didn't find a dedicated software tool (except for ModeRator (Mednis et al., 2012)) for comparison of models, there are several software tools with functionality that is more or less related to comparison of models. These include: Tools-4-Metatool (Xavier et al., 2011), Compare Subsystems (Oberhardt et al., 2011), SemanticSBML (Krause et al., 2010), COBRA (Becker et al., 2007), FAME (Boele et al., 2012) and MetRxn (Kumar et al., 2012).

The comparison of models starts at the comparison of metabolites. Chemical formulas (if available) and names (if available) of metabolites are the main features that can be compared. An algorithm for metabolite comparison analysing the similarity of chemical formula and name of metabolites is proposed. In case of identical formula and name the metabolites are considered to be identical automatically. In other cases the similarity rate is calculated and manual curation is needed for metabolites with the similarity above chosen threshold. The algorithm is tested comparing in pairs two models (or reconstructions) of E.coli and two of yeast (or S.cerevisiae).

## 2 Materials and methods

### 2.1. Comparison criteria

In order to compare two reactions, one has to compare the involved metabolites and thereby decide if two given reactions are equal. This means that before comparing reactions, one has to compare and map metabolites between both reconstructions. The mapping of metabolites means explicitly defining that metabolite “abc” from one reconstruction is the same as metabolite “xyz” from other reconstruction. Technically it can be achieved by assigning the same ID to two metabolites that are believed to be the same.

In a case that two reconstructions demanding a comparison come from different sources or authors, it is very likely that the elements in both reconstructions will not share common identifiers. In other words, elements, like compartments, metabolites and reactions will have different IDs. In SBML files there is unique id value for each element, but in COBRA models abbreviations serve as unique identifiers.

The problem is to evaluate and decide if two elements with different IDs are the same or not. If the IDs can not be used for identifying an element, some other property must be used. For metabolites such property could be a chemical formula. Chemical formulas in clear text (such as H2O) usually can be found in COBRA models where chemical formulas are mandatory data that enables some of the core functionality of COBRA toolbox. However, chemical formulas in SBML files are not a common practice. Even more -- the data model of SBML does not support chemical formulas in clear text. Chemical formulas as identifiers for metabolites cannot be used for one more reason -- the presence of isomers. The notation of chemical formula shows number of atoms in a molecule, and the same atoms with the same count but with different spatial structure are noted the same. Well known examples of isomers are glucose and fructose.

Other option is to compare elements, e. g. metabolites by names. Since the terminology is not yet standardized, it is very unlikely that different authors will name elements absolutely identically. Therefore a fuzzy string comparison algorithm can be used. Such an algorithm calculate similarity of two given text strings - metabolite names in our case.

Various implementations for calculating similarity ratio between two text strings are available.

A ratio usually is floating point number ranging from 0.0 to 1.0 indicating similarity of two given sequences (or strings). The ratio “0” means that two strings have nothing in common. For example such strings would be “ABCD” and “EFGH”. These particular strings do not share a single common character. The ratio “1.0” means that two given strings are absolutely identical. The closer to 1.0, the more similar two given strings are.

The distance between two strings is the number of steps needed to transform string A into string B. The distance sometimes is also called edit distance. For example the distance between “abcd” and “aZcZ” is 2. That's because there are only two edits needed to change “abcd” into “aZcZ” -- two replace operations.

The threshold is a measurable value that serves the purpose of filtering elements by a value of its property. For instance, threshold of 5, means that any value that is below (or above) 5 is filtered and not passed further in the algorithm. The threshold of string similarity ratio (and/or distance) means that all pairs of strings that are not similar enough, is filtered out.

The Table 1 shows examples of ratios and distances for different string pairs. Given examples show an interesting trend: string length have an impact on ratio. On longer strings the impact is smaller, but on shorter strings the impact is higher.

The authors of SBMLmerge sowtware (Schulz et al., 2006) are using only ratio (Ratcliff and Metzener, 1988) without taking into account edit distance to map metabolite names. As it is shown on Table 1 this approach works well only with long metabolite names. In order to find two similar names that are short in length, the threshold of ratio need to be lowered. However, lowering the threshold involves the higher risk false positives to be found. It means that on low threshold two names can be reported as similar, but actually they are two different metabolites, like “D-glutamate” and “L-glutamate”.

Table 1

**Example of Levenshtein ratio and edit distance variations for different strings**

| String A | String B | Ratio | Distance |
|---|---|---|---|
| Bicarbonate | bicarbonate | 0.9 | 1 |
| Glucose-6-phosphate | Glucose six phosphate | 0.8 | 5 |
| Glucose-6-phosphate | Glucose-six-phosphate | 0.9 | 3 |
| L-tryptophanyl-tRNAtrp | L-Tryptophanyl-tRNA(trp) | 0.86 | 4 |
| L-lysine | L-Lysine | 0.87 | 1 |
| D-glutamate | L-Glutamate | 0.81 | 2 |
| abcd | aZcZ | 0.5 | 2 |

### 2.2. Three level filtering

The comparison of metabolite names of large models can relate several metabolites of model M1 to one metabolite of model M2. The second level filters out multiple links to the model M1 (see Fig. 1). The third level filters out multiple links to the model M2. During first level filtering the strength of a link between all possible metabolite pairs between model M1 and M2 is calculated. During the third level filtering the relations "many to one" are compared and the pair with the highest similarity is nominated as pair of mapped metabolites.

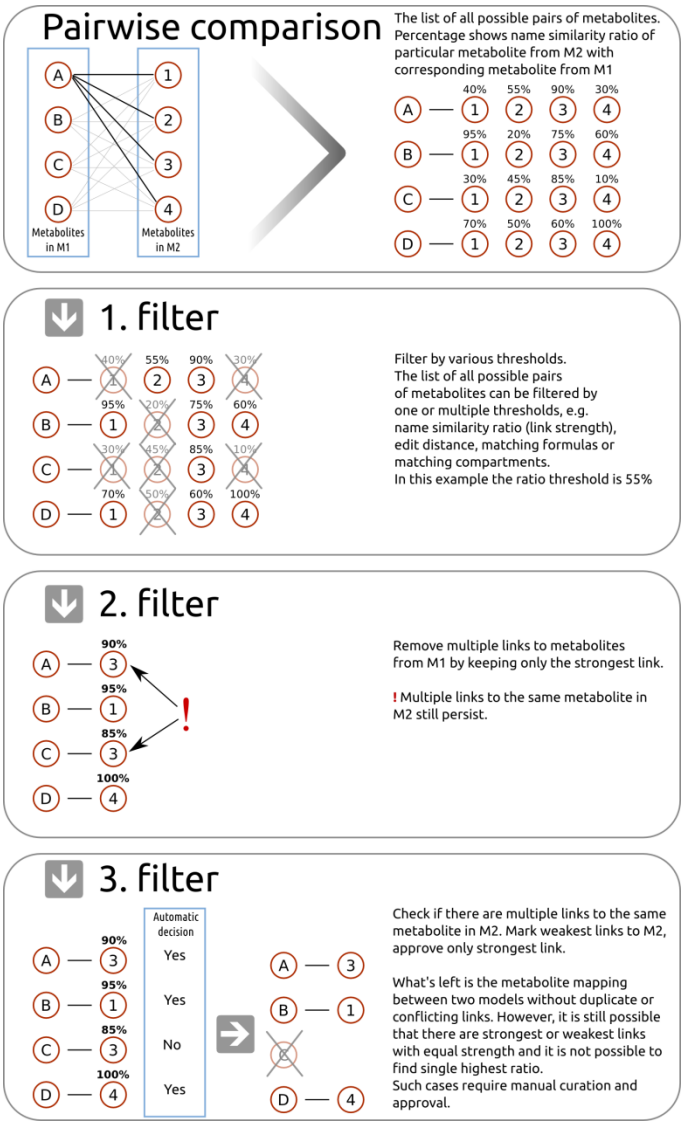


Fig. 1. **Schematic example of three level filtering approach. Metabolites of model M1 are marked by letters and metabolites of model M2 are marked by numbers. Name similarity ratio (strength of links) is measured in percents.**

The above mentioned approach is implemented the most recent (2.5.4) version of ModeRator - previously published software tool for model comparison (Mednis et al., 2012).

Two pairs of models were compared: *Escherichia coli* models "ecol199310cyc" and "ecol316407cyc" from BioCyc database (http://www.biocyc.org/) and *Saccharomyces cerevisiae* models iND750 developed by Natalie Duarte (Duarte et al., 2004) and iLL672 developed by Lars Kuepfer (Kuepfer et al., 2005).

## 3 Results and discussion

### 3.1. Comparison of E.coli models

E.coli models contain 1314 and 1704 metabolites (see the supplementary file "ecoli_metabolites"). Metabolites in both E.coli models were compared by identifiers, therefore filtering of multiple links was not applicable. Since E.coli models came from the same source, we expected to find equal annotation for same metabolites.

**Automatic comparison** using ModeRator revealed 1094 common metabolites with matching identifiers. Interestingly, that not all metabolite pairs with matching IDs were having equal annotation, particularly metabolite name and chemical formula. 31 of 1090 metabolites pairs were having different names. 30 pairs were having different chemical formula. For 391 metabolite pair it was not possible to compare chemical formulas, because one or both formulas were missing.

**Manual curation** of the above mentioned 31 pairs with non equal names was performed using databases (Ecocyc (Keseler et al., 2011), Metacyc (Caspi et al., 2012), and E. coli Metabolome database (Guo et al., 2012)). 30 metabolite pairs were approved during manual curation. One pair was left without decision.

### 3.2. Comparison of S.cerevisiae models

The reconstruction of Yeast metabolism iLL672 (Kuepfer et al., 2005) is based on a previous reconstuction iFF708 (Förster et al., 2003). iFF708 covered two main compartments, cytosol and mitochondria. Metabolites located in the mitochondria for a specific reaction terminate with an additional „m" (Förster et al., 2003). This differentiation between cytosolic and mitochondrial metabolites persists in iLL672 (Kuepfer et al., 2005). Metabolite localisation was ignored during manual comparisson (strings were matched ignoring the terminal „M"). The compared *Saccharomyces cerevisiae* models contain 679 and 1061 metabolites.

**Automatic comparison** by identifiers revealed only one metabolite - "Acyl-carrier protein", so we had to compare metabolites by names. From total of 447 returned results (pairs), 248 pairs with identical metabolite names (ratio 100%) were automatically approved. 199 pairs were with non 100% ratio match. During third level filtering 152 of them were automatically approved (considered as mapped metabolites

being equal) and 47 automatically disapproved (considered as being different metabolites).

Only Names of the metabolites were used for **manual curation**. For the manual comparisson of iLL672 and iND750 BIGG database (Schellenberger et al., 2010) was used. It captures detailed information about iND750 metabolites. Using this recource, many metabolite matches could be either confirmed or verified. In case the match could not be resolved we additionally queried the Yeast metabolome database (YMDB, http://www.ymdb.ca/(Jewison et al., 2012)). Using synonyms listed in YMDB, we again queried BIGG database and derived a number of alternative matches of iLL672 metabolites to iND750 metabolites (alternative matches provided in column „Comments" in supplementary file "yeast_metabolites").

In case a typing error was likely, match was confirmed and a note was made in column „Comments" (See supplementary materials file "yeast_metabolites").

Thresholds for name similarity ratio and edit distance were chosen low enough to minimize cases false negatives when two identical metabolites with not-so-identical names are filtered from results. The threshold for similarity ratio was 68% and the threshold for edit distance was 15 edits.

199 pairs with non 100% match (68 <= ratio (%) < 100) were curated. The decision of manual curation was: 133 decisions are the same as in automatic comparison, 64 are different and in two cases decision could not be made based on information in databases.

Many differences were found through excluding or including of „," ,"–„ or „spaces", and terms of stereoisomerism into metabolite names.

Table 2

**Manual curation vs. automatic comparison. „Disagree on SAME" means that manual curation decision is „SAME", but ModeRator decides otherwise.**

| Manual curation vs. automatic comparison | Cases | Comment |
|---|---|---|
| Identical names | 250 | Manual curation was not necessary |
| Agree on decision „SAME metabolites" | 124 | Manual and automatic curation decide that metabolites are the same |
| Agree on DIFFERENT | 42 | Manual and automatic curation decide that metabolites are different |
| Disagree on SAME | 9 | Manual curation decides that metabolites are the same |
| Disagree on DIFFERENT | 22 | Manual curation decides that metabolites are different |

## 4 Conclusion

Different formats and description standards of models as well as charged/uncharged formulas and different names and abbreviations of metabolites make comparison of reconstructions and models a complicated task.

In the software ModeRator implemented three level filtering approach can perform automatic metabolite matching as part of model comparison task. The three level filtering can be used as decision support system that processes the raw data to save time of manual curator. The algorithm can find identical metabolites which are declared as mapped metabolites without manual curation. The algorithm also filtrates away metabolite pairs with low similarity (levels of similarity thresholds can be adjusted) thus saving time of manual curation. Thus only third group of metabolite pairs with high similarity remains to be curated manually.

The two demonstrated application cases (pair of E.coli and pair of S.cerevisiae reconstructions) demonstrate that the success rate strongly depends on similarity of metabolite description approach used by model builders. In case of models built by the same scientific group automatic metabolite matching demonstrates very good performance and manual curation may be needed just for few metabolites.

## Acknowledgments

This work is funded by a project of European Structural Fund Nr. 2009/0207/1DP/1.1.1.2.0/09/APIA/VIAA/128 "Latvian Interdisciplinary Interuniversity Scientific Group of Systems Biology" www.sysbio.lv.

## References

Becker, S. a, Feist, A. M., Mo, M. L., Hannum, G., Palsson, B. Ø., and Herrgard, M. J. (2007). Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox. Nature protocols, 2(3), 727–38. doi:10.1038/nprot.2007.99

Boele, J., Olivier, B. G., and Teusink, B. (2012). FAME, the Flux Analysis and Modeling Environment. BMC systems biology, 6(1), 8. doi:10.1186/1752-0509-6-8

Caspi, R., Altman, T., Dreher, K., Fulcher, C. a, Subhraveti, P., Keseler, I. M., Kothari, A., et al. (2012). The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. Nucleic acids research, 40(Database issue), D742–53. doi:10.1093/nar/gkr1014

Duarte, N. C., Becker, S. a, Jamshidi, N., Thiele, I., Mo, M. L., Vo, T. D., Srivas, R., et al. (2007). Global reconstruction of the human metabolic network based on genomic and bibliomic data. Proceedings of the National Academy of Sciences of the United States of America, 104(6), 1777–82. doi:10.1073/pnas.0610772104

Duarte, N. C., Herrgård, M. J., and Palsson, B. Ø. (2004). Reconstruction and validation of Saccharomyces cerevisiae iND750, a fully compartmentalized genome-scale metabolic model. Genome research, 14(7), 1298–309. doi:10.1101/gr.2250904

Edwards, J. S., and Palsson, B. O. (1999). Systems properties of the Haemophilus influenzae Rd metabolic genotype. The Journal of Biological Chemistry, 274(25), 17410–17416. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/10364169

Finney, A., and Hucka, M. (2003). Systems biology markup language: Level 2 and beyond. Biochemical Society Transactions, 31(Pt 6), 1472–1473. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/14641091

Fleischmann, R. D., Adams, M. D., White, O., Clayton, R. A., Kirkness, E. F., Kerlavage, A. R., Bult, C. J., et al. (1995). Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. Science, 269(5223), 496–512. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/7542800

Förster, J., Famili, I., Fu, P., Palsson, B. Ø., and Nielsen, J. (2003). Genome-scale reconstruction of the Saccharomyces cerevisiae metabolic network. Genome research, 13(2), 244–53. doi:10.1101/gr.234503

Gay, S., Soliman, S., and Fages, F. (2010). A graphical method for reducing and relating models in systems biology. Bioinformatics, 26(18), i575–i581. Retrieved from http://www.bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btq388

Guo, a. C., Jewison, T., Wilson, M., Liu, Y., Knox, C., Djoumbou, Y., Lo, P., et al. (2012). ECMDB: The E. coli Metabolome Database. Nucleic Acids Research, (10), 1–6. doi:10.1093/nar/gks992

Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., Arkin, a. P., et al. (2003). The systems biology markup language (SBML): a medium for representation and exchange of biochemical network

models. Bioinformatics, 19(4), 524–531. doi:10.1093/bioinformatics/btg015

Jewison, T., Knox, C., Neveu, V., Djoumbou, Y., Guo, A. C., Lee, J., Liu, P., et al. (2012). YMDB: the Yeast Metabolome Database. Nucleic acids research, 40(Database issue), D815–20. doi:10.1093/nar/gkr916

Kanehisa, M., and Goto, S. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes. Nucleic Acids Research, 28(1), 27–30. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=102409&tool=pmcentrez&rendertype=abstract

Kanehisa, M., Goto, S., Sato, Y., Furumichi, M., and Tanabe, M. (2011). KEGG for integration and interpretation of large-scale molecular data sets. Nucleic Acids Research, 40(Database issue), D109–14. doi:10.1093/nar/gkr988

Karp, P. D., Ouzounis, C. a, Moore-Kochlacs, C., Goldovsky, L., Kaipa, P., Ahrén, D., Tsoka, S., et al. (2005). Expansion of the BioCyc collection of pathway/genome databases to 160 genomes. Nucleic acids research, 33(19), 6083–6089. doi:10.1093/nar/gki892

Karp, P. D., Paley, S. M., Krummenacker, M., Latendresse, M., Dale, J. M., Lee, T. J., Kaipa, P., et al. (2010). Pathway Tools version 13.0: integrated software for pathway/genome informatics and systems biology. Briefings in bioinformatics, 11(1), 40–79. doi:10.1093/bib/bbp043

Keseler, I. M., Collado-Vides, J., Santos-Zavaleta, A., Peralta-Gil, M., Gama-Castro, S., Muñiz-Rascado, L., Bonavides-Martinez, C., et al. (2011). EcoCyc: a comprehensive database of Escherichia coli biology. Nucleic acids research, 39(Database issue), D583–90. doi:10.1093/nar/gkq1143

Kostromins, A., and Stalidzans, E. (2012). Paint4Net: COBRA Toolbox extension for visualization of stoichiometric models of metabolism. Biosystems. doi:10.1016/j.biosystems.2012.03.002.

Krause, F., Uhlendorf, J., Lubitz, T., Schulz, M., Klipp, E., and Liebermeister, W. (2010). Annotation and merging of SBML models with semanticSBML. Bioinformatics (Oxford, England), 26(3), 421–2. doi:10.1093/bioinformatics/btp642

Kuepfer, L., Sauer, U., and Blank, L. M. (2005). Metabolic functions of duplicate genes in Saccharomyces cerevisiae. Genome research, 15(10), 1421–30. doi:10.1101/gr.3992505

Kumar, A., Suthers, P. F., and Maranas, C. D. (2012). MetRxn: a knowledgebase of metabolites and reactions spanning metabolic models and databases. BMC bioinformatics, 13(1), 6. doi:10.1186/1471-2105-13-6

Lewis, N. E., Jamshidi, N., Thiele, I., and Palsson, B. Ø. (2009). Metabolic systems biology: a constraint-based approach. In R. Meyers (Ed.), Encyclopedia of Complexity and Systems Science (p. 5535). New York: Springer.

Mednis, M., Rove, Z., and Galvanauskas, V. (2012). ModeRator - a software tool for comparison of stoichiometric models. 7th IEEE International Symposium on Applied Computational Intelligence and Informatics (pp. 97–100).

Oberhardt, M. A., Puchałka, J., Martins Dos Santos, V. A. P., and Papin, J. A. (2011). Reconciliation of Genome-Scale Metabolic Reconstructions for Comparative Systems Analysis. (P. E. Bourne, Ed.)PLoS Computational Biology, 7(3), 18. Retrieved from http://dx.plos.org/10.1371/journal.pcbi.1001116

Palsson, B. Ø. (2006). Systems Biology: Properties of reconstructed networks. Cambridge University Press.

Radulescu, O., Gorban, A. N., Zinovyev, A., and Lilienbaum, A. (2008). Robust simplifications of multiscale biochemical networks. BMC systems biology, 2(1), 86. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2654786&tool=pmcentrez&rendertype=abstract

Ratcliff, J. W., and Metzener, D. (1988). Pattern Matching: The Gestalt Approach. Dr Dobbs Journal, (July), 46.

Rubina, T., and Stalidzans, E. (2010). Topological features and parameters of Biochemical Network Structures. International Industrial Simulation Conference (pp. 228–236). Budapest: EUROSIS.

Schellenberger, J., Park, J. O., Conrad, T. M., and Palsson, B. Ø. (2010). BiGG: a Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions. BMC bioinformatics, 11, 213. doi:10.1186/1471-2105-11-213

Schellenberger, J., Que, R., Fleming, R. M. T., Thiele, I., Orth, J. D., Feist, A. M., Zielinski, D. C., et al. (2011). Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2.0. Nature Protocols, 6(9), 1290–1307. doi:10.1038/nprot.2011.308

Schulz, M., Uhlendorf, J., Klipp, E., and Liebermeister, W. (2006). SBMLmerge, a system for combining biochemical network models. Genome informatics International Conference on Genome Informatics, 17(1), 62–71. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/17503356

Thiele, I., and Palsson, B. Ø. (2010). A protocol for generating a high-quality genome-scale metabolic reconstruction. Nature protocols, 5(1), 93–121. doi:10.1038/nprot.2009.203

Whitaker, J. W., Letunic, I., McConkey, G. A., and Westhead, D. R. (2009). metaTIGER: a metabolic evolution resource. Nucleic Acids Research, 37(Database issue), D531–D538. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2686446&tool=pmcentrez&rendertype=abstract

Xavier, D., Vázquez, S., Higuera, C., Morán, F., and Montero, F. (2011). Tools-4-Metatool (T4M): Online suite of web-tools to process stoichiometric network analysis data from METATOOL. Biosystems, 1–4. doi:10.1016/j.biosystems.2011.04.004

Yamada, T., and Bork, P. (2009). Evolution of biomolecular networks: lessons from metabolic and protein interactions. Nature reviews. Molecular cell biology, 10(11), 791–803. doi:10.1038/nrm2787

# Individual tree identification using different LIDAR and optical imagery data processing methods

**Ingus Smits[1,2*], Gints Prieditis[2], Salvis Dagis[1,2], Dagnis Dubrovskis[1]**

[1]*Precision Forestry Research Group, Forest faculty, Latvia University of Agriculture, Akademijas iela 11, LV3001, Jelgava, Latvia*
[2]*Faculty of Information Technologies, Latvia University of Agriculture, Liela iela 2, LV3001, Jelgava, Latvia*
**Corresponding author*
*Ingus.smits@gmail.com*

Received: 25 November 2012; accepted: 28 November 2012; published online: 29 November 2012.
This paper has no supplementary material.

**Abstract:** *The most important part in forest inventory based on remote sensing data is individual tree identification, because only when the tree is identified, we can try to determine its characteristic features. The objective of research is to explore remote sensing methods to determine individual tree position using LIDAR and digital aerial photography in Latvian forest conditions. The study site is a forest in the middle of Latvia at Jelgava district (56°39' N, 23°47' E). Aerial photography camera (ADS 40) and laser scanner (ALS 50 II) were used to capture the data. A LIDAR data is 1.4 to 9 p/m2 depending on the altitude. Image data is RGB (Red, Green, and Blue), NIR (Near Infrared) and PAN (Panchromatic) spectrum with 20 to 50 cm pixel resolution depending on the altitude. Image processing was made using Fourier transform and RGB colour segmentation. LIDAR data processing methods were DBSCAN algorithm, global maximum algorithm, and local maximum algorithm. Field measurements were tree coordinates, species, height, diameter at breast height, crown width, and length.*
*Best results on both ALS and ADS data were achieved using local maximum methods.*

**Keywords:** Forest inventory, tree identification, laser scanning, aerial photography, data fusion.

## 1. Introduction

The most responsible and important part in forest inventory based on remote sensing data is individual tree identification, because only when the tree is identified, we can try to determine its characteristic features, like tree species, tree height, diameter at breast height, volume, and biomass (Secord et al., 2006; Edson and Wing, 2011).

In the studies of forest inventory using remote sensing sensors, one of the main problems that the authors mentioned is tree identification and tree location accurate determination (Hyyppä et al., 2008; Kane et al., 2010), especially in Middle Europe (Diedershagen et al., 2006), since there is a mixture of different deciduous and coniferous trees. As a result, the identification is more difficult. Many authors in their conclusions highlight that the usage of LIDAR and airphoto methods to determine forest inventory parameters will never be one hundred per cent correct (Onge et al., 2004; Rombouts, 2006), especially applying automated tracking methods (Hyyppä et al., 2004; Junttila et al., 2010). Practically for all researchers, so far it has been difficult to identify small trees (Pitkänen, 2001; Pouliot and King, 2005) and closely growing trees (Pouliot and King, 2005; Koch et al., 2006), as well as high density hardwood stands with homogeneous crown (Koch et al., 2006; Rahman and Gorte, 2008). Automated tree identification and tree location accurate determination are still problematic (Popescu et al., 2002; Junttila et al., 2010), even in cases where access to different types of data (Vauhkonen et al., 2008) is available. This is mainly by the fact that trees vary in crown size (Tokola et al., 2008), shape and optical properties (Tokola et al., 2008; Vauhkonen et al., 2008). For example, some species have rounded crowns, some have cone-shaped crowns, and some have star-shaped crowns. Tone in aerial photographs depends on many factors, and relative tones on a single photograph, or a strip of photographs may be of a great value in delineating adjacent trees of different species (Koch et al., 2006). Crowns are often interlaced. Occlusion and shading are present and result in omission errors. These factors affect the treetop positioning and make the identification of trees difficult.

Numbers of different methods are used to identify a single tree. The main criterion for choice of identification method is the specific structure of forest canopy and species diversity. If the area of construction is more complicated, tree locations and their exact coordinates are difficult to determine. Single-scale template matching has been successfully applied in 2D and 3D treetop estimation of regular stands, where crowns show only moderate variation (Korpela, 2006). In contrast, to determine all the treetops where forest foliage is complex in structure and with a large variation, the most appropriate are the automatic and semi-automatic methods (Korpela et al., 2007).

Pitkanen developed several methods for individual tree detection based on canopy height model of Airborne LIDAR. In one of them, he used a Gaussian filter to determine equalized height of pixel and local maxima on the smoothed Canopy Height Model were considered as tree locations. In the other method, large numbers of possible tree locations were selected based on local maxima. The pixels were reduced based on the slope within the assumed crown centre area and based on the distance and valley depth between a location and its neighbouring locations. The second method used crown width and tree height model as a parameter to adapt with tree

size. Both methods showed that about 60- 70% of the dominant trees were found (Pitkanen et al., 2004).

Heinzel used local maximum of smoothed canopy height model and delineation of single tree is done using pouring algorithm. It was observed that the segmented trees still contained a lot of wrong segments, in which the regions are too small to be a tree, inappropriate crown shape, and crown regions that cover another trees and canopy gaps. The segments were refined based on their shapes and distance between tree tops (Heinzel et al., 2008).

Data collection and processing methods at different conditions work differently, mainly due to forest density, represented tree species and forest diversity in growing conditions, as well as LIDAR and digital aerial cameras technology specifics.

The objective of research is to explore methods to determine single tree position using LIDAR and digital aerial photography in Latvian forest conditions.

## 2. Materials and methods

The study site is a forest in the middle of Latvia in Jelgava district (56º39' N, 23º47' E). The area consists of mixed coniferous and deciduous forest with different age, high density, complex structure, various components, and composition. Represented species are pine (*Pinus sylvestris L.*), spruce (*Picea abies (L.) H.Karst*), birch (*Betula pendula Roth*), and aspen (*Populus tremula L.*).

Data were obtained using a specialized aircraft Pilatus PC-6, which is equipped with a positioning and Geomatics technology company Leica Geosystems equipment - a large format digital aerial photography camera (ADS 40) and laser scanner (ALS 50 II). The study area was flown over by plane and scanned at three different altitudes. A LIDAR digital terrain models (DTM) were estimated from leaf-on data from May, 2010 having 9 p/m2 at 500 m altitude; 3.7 p/m2 at 1000 m altitude; 1.4 p/m2 at 1500 m altitude.

The image data is RGB (Red, Green, and Blue), NIR (Near Infrared) and PAN (Panchromatic) spectrum with 20 cm pixel resolution at 500 m altitude; 30 cm pixel resolution at 1000 m altitude; 50 cm pixel resolution at 1500 m altitude.

In the study area, 10 sample plots were selected to analyze accuracy of different tree identification methods and to determine impact of the resulting ALS and ADS data structures on the number of trees identified. Plots were chosen to be simple by the structure with small proportion of the tree on the second floor.

In order to determine the best method ALS data with 9 p/m2 and ADS data with a pixel size of 20 cm in the field were used.

It should be noted that the choice of methods was based only on the number of trees identified in all the plots together, without analyzing them over the tree species or forest floors or other woodwork characterizing parameters.

All trees with a diameter at breast height DBH of more than 5 cm were measured and for each tree coordinates, its species, height, DBH, crown width and length were recorded. Altogether there were 252 trees in the data. Circular sample plots were established with dimensions of 0.045 ha.

Differentially corrected Global Positioning System measurements were used to determine the position of each plot centre. The accuracy of the positioning was approximately 1 meter.

### 2.1. LIDAR data processing methods

Three different methods for tree identification were evaluated. First method is based on reflection point count in a certain height range. It was made by adopting density based clustering algorithm (DBSCAN) (Sriperumbudur and Steinwart, 2012; Meng, 2010), which was accompanied by restrictions on the radius determination. Realization of the method is based on the number of points above a certain height level (Fig. 1.).

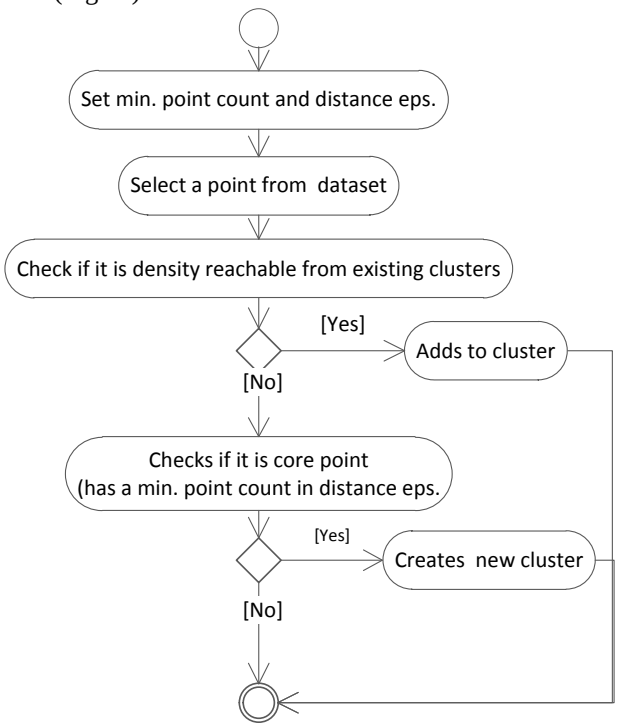


Fig. 1. **DBSCAN algorithm** (Sriperumbudur and Steinwart, 2012; Meng, 2010)

In the literature, several researchers state that this method gives good results (Meng, 2010; Sriperumbudur and Steinwart, 2012).

The second method used for processing of LIDAR data set is global maximum method (Fig. 2.) that uses height data and range limitations. This method worked poorly. First, the LIDAR data set points were read, and then divided into quadrants. Afterwards cluster formed by the maximum points in the upper layer was found, and deleted from the cube. In this way, part of points belonging to other trees was lost, and trees were omitted.

The third method used was searching for local maximums on height axis of LIDAR data collection. Use of this method is based on the assumption that tree top centre is highest point in data set which is not always the case. This method is used on LIDAR data that are smoothed by using Gaussian mask. As closest point of such mask has bigger affect than the ones on the border. It can be stated that this filter evaluates between point interactions. After calculating the Gaussian mask the highest segment points above the surface were searched and compared with adjacent cells independently each segment. If the selected cell is higher than the adjacent - then there is the tree top. Tree top is not always the centre of the cell, so the tree is found in the centre of determining the highest cell. Tree recognition algorithm is shown in figure (Fig. 3.)

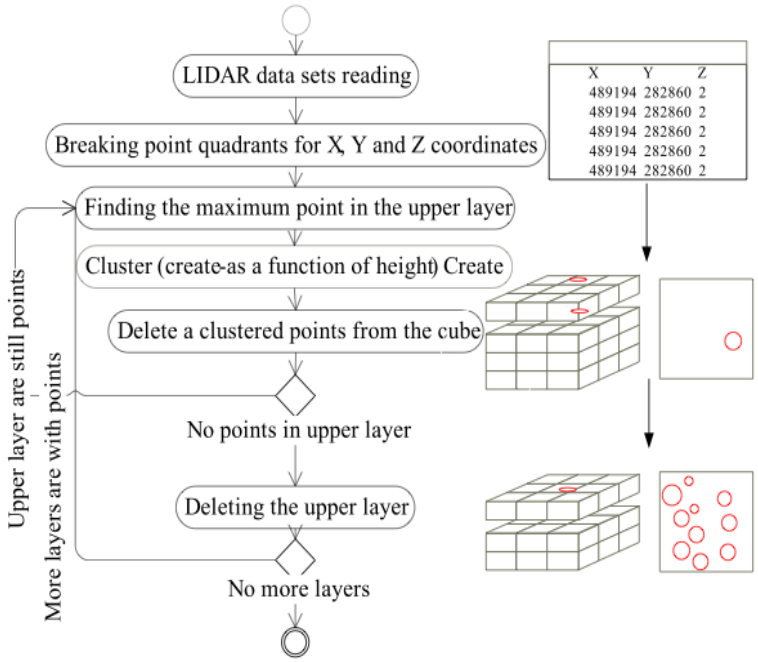


Fig. 2. **Global maximum algorithm**

The described local maximum approach is one of the most widely used methods of tree identification, and determination of the crown canopy tree height determination (Pikanen et al., 2004; Popescu, 2003; Korpela, 2006; Korpela, 2007).

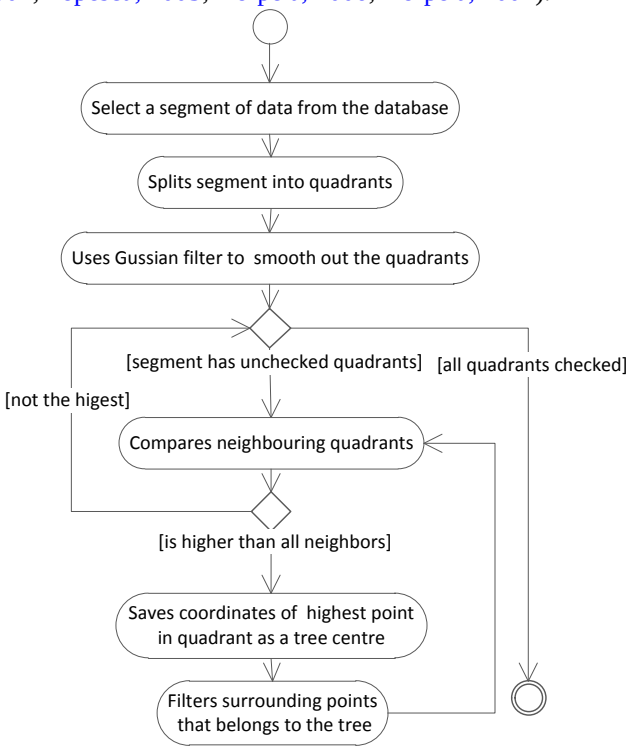


Fig. 3. **Local maximum algorithm**

In the course of evaluating the capability of identifying trees using ALS data, all three methods were examined, and for future use on the local maximum approach and the Gaussian filter were chosen as these approaches showed the best result in practical sample plot tests.

## 2.2. Image processing methods

RGB colour segmentation is one of the most commonly used image processing methods (Crosilla, et al., 2005), and in this research it was considered as an alternative to the local maximum approach (described later). Two phase process that consists of image preparing and processing steps was realized for tree identification using a segmentation method. In the image preparatory phase, smaller images from the aerial photographs were created and geographical information for each of them was stored. In segmentation process, each image was divided into several regions. Result of this process is a set of segments, covering the entire image or individual object contours, that can be facilitated in further image analysis and processing tasks. All pixels in the segment have common colour, intensity, texture and other characteristics. For this study, colour segmentation algorithm that filters certain colour values specified by predefined masks was used on images green channels data.

For image processing, different algorithms can be used to improve or on the contrary - to lower quality in order to avoid some noise in the data. Still by using standard image processing algorithms it is impossible to do an automated identification. For tree identification from aerial images it is necessary to develop special algorithms and methods that are based or use classical ones in some detailed tasks. Such method can be seen in figure (Fig. 4.).

In this method (Fig. 4), tree identification process begins with detection of a pixel that belongs to the tree canopy. Once one pixel is found in given direction, algorithm looks for the other side of segment (border pixels). In a next step, centre of the vector given by two points is found and used to search the other border points in different directions that help to establish a correct tree top centre position. Geographic coordinates for identified trees are calculated using image geo-referenced data and its pixels are excluded from the searched area. After determination of tree centre coordinates, they are imported into the database. In addition to the coordinate information, the image colour component values and all other available data are recorded for postprocessing tasks.

Tree identification through segmentation method is less labour intensive as compared to the local maximum method, as

well as less complex from calculation point of view. But in the practical tests on selected sample plots it shows worse results than the local maximum method, and therefore it was not used for all sample plots in this study.

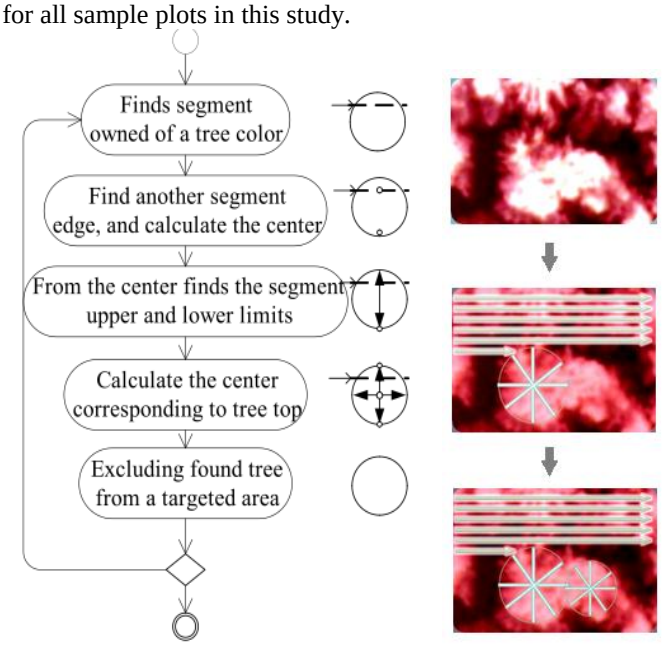


Fig. 4. **Tree colour segmentation method**

The main method used in this study for tree identification from aerial photographs is based on the local maximum approach (Rossmann, et al., 2007; Popescu and Wynne, 2002), where using the Fourier transform process that consists of several stages- image preparation, image processing and compilation of results- is performed.

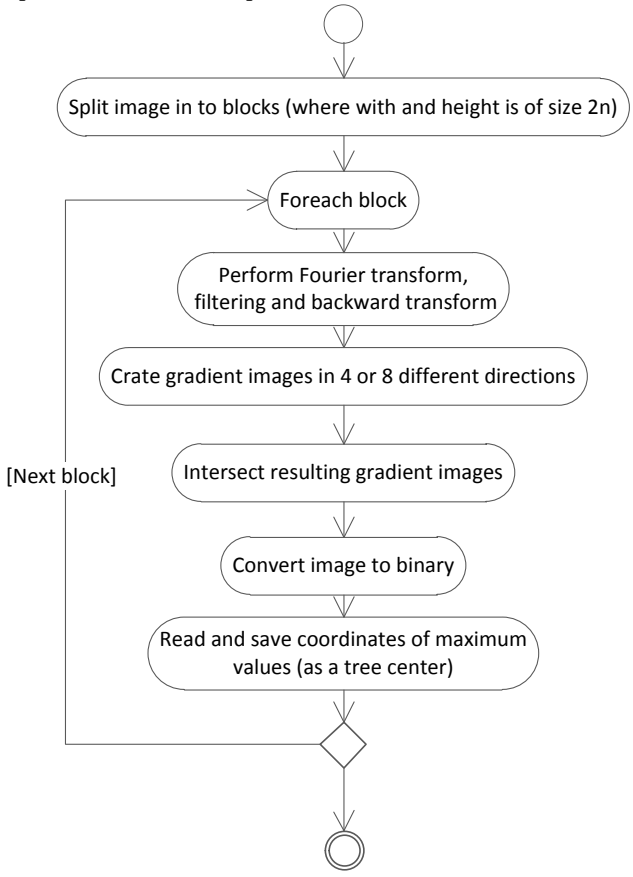


Fig. 5. **Local maximum method for ADS data**

Figure (Fig. 5) shows main steps used in local maximum method for tree identification. It begins with image division into several sub images of size 2n. Main reason for such deviation and size restrictions is dictated by fast Fourier transformation algorithm used in next steps. After Fourier transformation each subimage is filtered and transformed back to spatial domain. In a next step gradient images in 4 or 8 different directions are calculated and intersected with each other. Then the binary image is created, by using results of previous steps such that maximal values show only local maximum points. Main reason for choosing Fourier transformation and performing filtering in frequency domain instead of simple Gaussian filtering is speed of used methods and descriptions of successful usage found in publications. Fourier transformation is described as a method of choice for tree identification (Vaughn et al., 2011; Vaughn et al., 2012; Edwards and Nesbitt, 2002), and it is also tested in tree species identification tasks (; Nicholas et al., 2012).

Usage of Fourier transformation is studied both for tree position (Vaughn et al., 2011; Vaughn et al., 2012; Edwards and Nesbitt, 2002), and species identification (Nicholas et al., 2012).

## 3. Results and discussion

### 3.1. Comparison of tree identification methods used in the study

Before the remote sensing data processing 10 sample plots in the study area were selected for compliance evaluation of different tree identification methods, as well as for identification of ALS and ADS data collection altitude affects on the tree identification process. Totally 252 trees were measured in sample plots. Plots were chosen to be structurally simple, so the proportion of second floor trees is as small as possible. At first, the most accurate methods using the ALS data with 9 p/m2 and ADS image resolution with a pixel size of 20 cm in the nature were established. Results of tree identification methods employed for comparison are shown in figure (Fig. 6.).

Then their results were evaluated on data gathered at different altitude. The results can be seen in figures (Fig. 7., 8.) It should be noted that the comparison is based only on the number of trees identified in all sample plots together, without analyzing them over the tree species or forest floors or other woodwork characterizing parameters. Local maximum method with a Gaussian filter for ALS data and Fourier filter for ADS data were considered to be the most accurate method for identifying the trees. Consequently these methods were used for tasks of tree identification for all sample plots in the study area.

Number of researchers have successfully used DB SCAN algorithm, but in practical sample plot test the algorithm showed poor results.

Lack of precision of the global maximum method may be explained by the fact that part of the ALS data set points, after finding global maximum, is attached to a single tree and removed from future search. In this case, each wrongly deleted point can lead to some omission errors.

Weak results of tree colour segmentation algorithm of image data could be explained by the fact that several trees are considered to be as one which leads to incorrect tree count results.

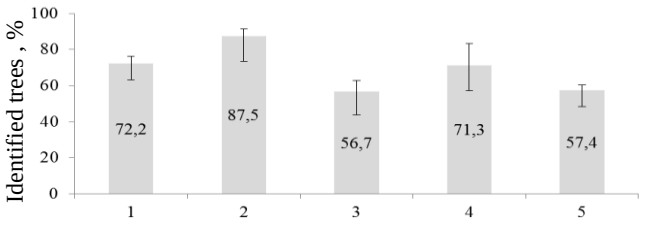


Fig 6. **Comparison of methods for tree identification. 1- Global maximum method (ALS data used); 2 – Local maximum method with Gaussian mask (ALS data used); 3 – DB SCAN algorithm ( ALS data used); 4 – Local maximum method with Fourier filtering (ADS data used); 5 – Tree colour segmentation method (ADS data). ALS data with 9 p/m2 and ADS image resolution with a pixel size of 20 cm in the nature used. Deviation intervals show the minimum and maximum values.**

Tree identification process is one of the most important stages in forest inventory, which is based on a separate survey of trees from remote sensing data, because only when a tree is identified, it is possible to perform other measurements and make predictions about forest characteristics.

## 3.2. Evaluation of results acquired from remote sensing data of different heights

ALS point density per square meter depends on flight altitude. So the data at different heights is processed to evaluate the effect of ALS point density change on the tree identification outcome. Figure (Fig. 7.) shows percent of identified trees at different point densities.

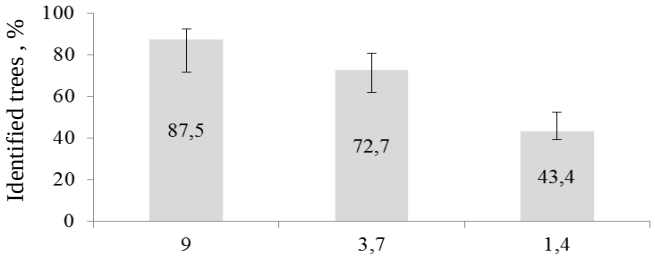


Fig. 7. **Identified trees at different point densities. The evaluation was carried out in 10 sample plots of study area (252 trees). Plots were chosen to be structurally simple, so the proportion of second floor trees is as small as possible. Deviation intervals show the minimum and maximum values.**

Best results of tree identification process are reached at the highest point density per square meter, which would be understandable, but the most interesting part is that ALS data with average point density also shows fairly good results. Although in the study for data analysis and processing mostly data with highest density were used, this evaluation shows that ALS data with average point density could be used in practice if needed.

The same as ALS data ADS image resolution depends on the flight altitude ADS. Figure (Fig. 8.) shows impact of ADS resolution change (result of changing flight altitude) on results of tree identification process.

Similarly as with ALS data, the best tree identification results are achieved with higher resolution ADS images. Equivalent results show that a medium-resolution aerial photographs give fairly good results. For data analysis and processing of aerial photographs in this study, the images with 20 cm pixel size in nature were used.

In order to improve the process of tree identification, ALS and ADS data were combined. As a result, the number of identified trees in the best situation is representative of 92.3%. Possible options for ALS and ADS data aggregation and the results are shown in figure (Fig. 9).

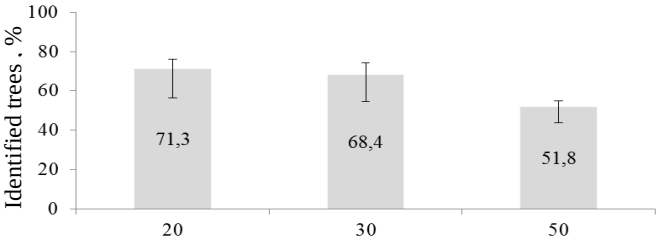


Fig. 8. **Identified trees at different pixel size. The evaluation was carried out in 10 sample plots of study area (252 trees). Plots were chosen to be structurally simple, so the proportion of second floor trees is as small as possible. Deviation intervals show the minimum and maximum values.**

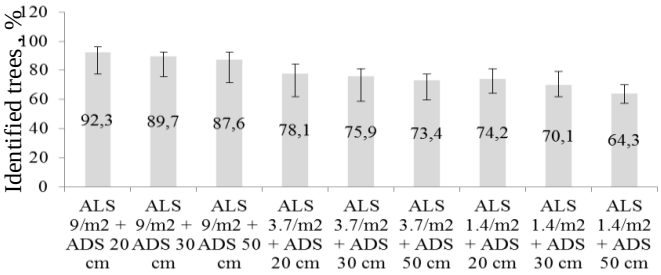


Fig. 9. **ALS and ADS data aggregation results. The evaluation was carried out in 10 sample plots of study area (252 trees). Plots were chosen to be structurally simple, so the proportion of second floor trees is as small as possible. Deviation intervals show the minimum and maximum values.**

The combined methods show better results because a part of the trees, which are not recognized by the first method, will be recognized by the other and vice versa. ADS and ALS data consolidation result is shown in figure (Fig.10.).

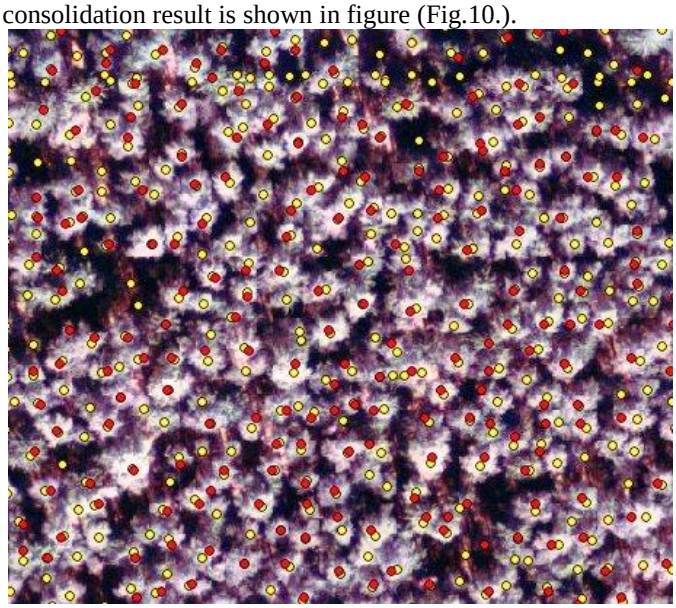

Fig. 10. **Trees identified in aggregated ALS and ADS data. The illustration shows an orthophoto of the study area. The red points show the trees identified using ALS data processing methods, and the yellow points show the trees identified using ALS data processing methods. The flight altitude was 500 m, 9 ALS and ADS p/m2 20 cm pixels in nature.**

## 4. Conclusion

Local maximum method with a Gaussian filter for ALS data and Fourier filter for ADS data showed the best results in practical sample plot tests and were considered the best for practical use in Latvian conditions.

Results of tree identification process can be improved by merging ALS and ADS results.

ALS and ADS data structure has a significant impact on the number of identified trees. More trees can be identified with higher resolution ADS and with a higher point density ALS data.

Latvian forest conditions are difficult for single tree remote sensing methods mainly due to mixed deciduous and coniferous spaces with high level of the second storey trees in one stand. Mostly the trees are close to each other, with high density and homogeneous crown. That is one of the main reasons for a large number of trees that are omitted.

Number of recognized trees could be improved by performing laser scanning in spring when the forest is less dense, the first storey trees are more transparent and the smaller-dimension trees can be recognized. Also tree crown shape analyse from LIDAR data can be used, and it means that there is a need for LIDAR data with a higher level of point density per square meter.

## References


Crosilla, F., Visintini, D., Sepic, F. (2005). A segmentation procedure of LiDAR data by applying mixed parametric and nonparametric models. In: Proceedings of the ISPRS Workshop Laser scanning 2005, ISPRS Archives, Vol. 36: conference paper. Enschede (Netherlands), September 2005, p. 132-137.

Diedershagen, O., Koch, B., Weinacker, H. (2006). Automatic Estimation of Forest Inventory Parameters Based on LIDAR, Multi-spectral and Fogis Data. Department of Remote Sensing and Land Information Systems. Institute of Forestry Economics. University Freiburg, Germany, 10 p.

Edson, C. and Wing, M.G. (2011). Airborne Light Detection and Ranging (LiDAR) for Individual Tree Stem Location, Height, and Biomass Measurements. Remote Sensing, 3, pp. 2494 – 2528.

Edwards, H.G.M., de Oliveira, L.F.C., Nesbitt, M. (2002). Fourier-transform Raman characterization of brazilwood trees and substitutes. The Analyst, Vol. 128, Nr. 1, 25 November 2002, p. 82 - 87. ISSN 1364-5528 (web), ISSN 0003-2654 (print)

Heinzel, J. N., Weinacker, H., Koch, B.(2008). Full automatic detection of tree species based on delineated single tree crowns - a data fusion approach for airborne laser scanning data and aerial photographs. In: SilviLaser 2008 Proceedings: conference proceedings. Edinburgh (UK), 2008, p. 76. – 85.

Hyyppä, J., Hyyppä, H., Leckie, D. (2008). Review of Methods of Small-footprint Airborne Laser Scanning for Extracting Forest Inventory Data in Boreal Forests. International Journal of Remote Sensing, 29, pp. 339-366.

Hyyppä, J., Hyyppä, H., Litkey, P. (2004). Algorithms and Methods of Airborne Laser Scanning for Forest Measurements. Remote Sensing and Spatial Information Sciences, 36, pp. 18-25.

Junttila, V., Kauranne, T., Leppänen, V. (2010). Estimation of Forest Stand Parameters from Airborne Laser Scanning Using Calibrated Plot Databases. Forest Science, 56, pp. 257-270.

Kane, V.R., Bakker, J.D., McGaughey, R.J., Lutz, J.A., Gersonde, R.F., Franklin, J.F. (2010). Examining conifer canopy structural complexity across forest ages and elevations with LiDAR data. Canadian Journal of Forest Research, 40, pp. 774-787.

Koch, B., Heyder, U., Weinacker, H. (2006). Detection of Individual Tree Crowns in Airborne Lidar Data. Photogrammetric Engineering & Remote Sensing, 72, pp. 357–363.

Korpela, I. (2006). Incorporation of Allometry into Single-tree Remote Sensing with LIDAR and Multiple Areal Images. Department of Forest Resource Management. University of Helsinki, Finland. p. 6.

Korpela, I., Dahlin, B., Schäfe, H. (2007). Single-tree forest inventory using LIDAR and areal images for 3D treetop positioning, species recognition, height and crown widh estimation. Department of Forest Resource Management. University of Helsinki, Finland. Available at: http://www.isprs.org/proceedings/XXXVI/3-W52/final_papers/Korpela_2007.pdf , 16 January 2009.

Korpela, I., Tokola, T., Ørka, H.O., Koskinen, M. (2004). Small – Footprint Discrete – Return LiDAR in Tree Species Recognition. Environmental Sciences, 387, pp. 1381-1386.

Korpela, I. and Tokola, T.E. (2006). Potential of Aerial Image-Based Monoscopic and Multiview Single-Tree Forest Inventory: A Simulation Approach. Forest Science, 52, pp. 136-147.

Meng, H., Currit, N., Zhao, K. (2010). Ground Filtering Algorithms for Airborne LiDAR Data: A Review of Critical Issues. Remote Sensing, Vol.2, 22 March 2010, p. 833 – 860. ISSN 2072-4292

Nicholas, R., Vaughn, L., Moskal, M. and Eric, C. (2012). Turnblom.Tree Species Detection Accuracies Using Discrete Point Lidar and Airborne Waveform Lidar. Remote Sensing. Vol. 4, 2012, p. 377-403. ISSN 2072-4292.

Onge, B., Jumelet, J., Cobello, M. (2004). Measuring Individual Tree Height Using a Combination of Stereophotogrammetry and Lidar. Canadian Jornal of Forest Research, 34, pp. 22–30.

Pitkänen, J. (2001). Individual tree detection in digital aerial images by combining locally adaptive binarization and local maxima methods. Canadian Journal of Forest Research, 31, pp. 832 – 844.

Pitkänen, J., Maltamo, M., Hyyppäb, J., Yub, X. (2004). Adaptive Methods for Individual Tree Detection on Airborne Laser Based Canopy Height Model. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 36 – 8/W2, pp. 187 – 191.

Popescu, S.C., Wynne, R.H., Nelson, R.F. (2002). Estimating plot-level tree heights with lidar: local filtering with a canopy-height based variable window size. Computers and Electronics in Agriculture, Vol. 37, Nr. 1 – 3, Dec 2002, p. 71 – 95. ISSN: 0168-1699.

Pouliot, D. and King, D. (2005). Approaches for optimal automated individual tree crown detection in regenerating coniferous forests. Canadian Journal of Remote Sensing, 31, pp. 255 – 267.

Rahman, M.Z.A., Gorte, B. (2008). Individual Tree Detection Based on Densities of Heigh Points of Heigh Resolution Airborne LiDAR. Available at: http://www.isprs.org/proceedings/XXXVIII/4-C1/Sessions/Session12/6790_Rahman_Proc.pdf, 4 January 2012.

Rombouts, J. (2006). Application of airbone LIDAR in forestry in North America and Scandinavia. The National Educational Trust of the Australian Forest Products Industries Fund. Australia. 2006. Available at: http://www.gottsteintrust.org/media/jrombouts.pdf, 16 January 2009.

Rossmann, J., Schluse, M., Bücken, A., Krahwinkler, P. (2007). Using Airborne Laser-Scanner-Data in Forestry Management: a Novel Approach to Single Tree Delineation. In: ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007: conference proceedings. Vol. 36, Part 3/W52. Espoo (Finland), 2007, p. 350 – 354.

Secord, J., Zakhor, (2010). A. Tree Detection in Aerial LiDAR and Image Data [online]. [s.a.] [cited 23 March 2010]. Available: http://www-video.eecs.berkeley.edu/papers/JSecord/ICIP2006_secord.pdf

Sriperumbudur, B.K., Steinwart, I. (2012). Consistency and rates for clustering with DBSCAN. In: AISTATS 2012 (International Conference on Artificial Intelligence and Statistics: conference paper. La Palma (Spain), April 2012, p. 1090 – 1098.

Tokola, T., Vauhkonen, J., Leppänen, V., Pusa, T., Mehtätalo, L., Pitkänen, J. (2008). Applied 3D Texture Features in ALS – Based Tree Species Segmentation. Available at: http://www.isprs.org/proceedings/XXXVIII/4-C1/Sessions/Session12/6724_Tokola_Proc.pdf, 4 January 2012.

Vaughn, N.R., Moskal, L.M., Turnblom, E.C. (2011). Fourier transformation of waveform Lidar for species recognition. Remote Sensing Letters, Vol. 2, Nr. 4, December 2011, p. 347 – 356. ISSN 2150-7058.

Vaughn, N.R., Moskal, L.M., Turnblom, E.C.(2012). Tree Species Detection Accuracies Using Discrete Point Lidar and Airborne Waveform Lidar. Remote Sensing, Vol. 4, Nr. 2, 2 February 2012, p. 377 – 403. ISSN 2072-4292.

Vauhkonen, J., Tokola, T., Leppänen, V. (2008). Applied 3D texture features in ALS-based tree species segmentation. University of Joensuu, Faculty of Forest Sciences. Available at: http://homepages.ucalgary.ca/~gjhay/geobia/Proceedings/Sessions/Session12/6724_Tokola_Proc.pdf , 16 January 2009.

# Tools for analysis of biochemical network topology

**Tatjana Rubina**[1*]

[1]*Department of Computer Systems, Latvia University of Agriculture, Liela iela 2, LV3001, Jelgava, Latvia*
**Corresponding author*
*Tatjana.rubina@llu.lv*

Received: 4 November 2012; accepted: 12 November 2012; published online: 13 November 2012.
This paper has no supplementary material.

**Abstract:** *The biochemical networks can present the relationships between genes and gene products, proteins, metabolites and etc. The exploration of these networks helps to understand cellular processes, functions or properties of biological system. The growing size of interaction models of biological system building elements request determination of the most important topological measurements of given task and powerful automated software tools to perform the analysis.*
*The network structure measures and properties are categorized in five groups: topological parameters, topological features, network metrics, network motifs and quantitative parameters of whole network structure. Topology analysis related features of software tools Cytoscape with plug-ins BiNoM and NetworkAnalyzer, VisANT, Biological Networks and CelNetAnalyser are reviewed to simplify the task-dependent choice. The applicability of software tools for calculation of 44 topological features is summarized.*
*Research resulted in overview on biochemical network structure analysis, on used topological features with the following goals: 1) to accumulate the existing knowledge about the network structure analysis; 2) to provide a list of topological parameters and features; 3) to provide the information of the existing software tools for the structure analysis.*

**Keywords:** Network, structural model, graph theory, computer software.

## 1. Introduction

The number of biological experiments, corresponding data sets and discoveries in postgenomic era have been very intensive (Gehlenborg et al., 2010; Ideker et al., 2001). One of directions of fast data generation has been systems biology which is concentrating on study of biological interaction networks (Durek and Walther, 2008; Zhu et al., 2007). Cellular proliferation, differentiation, and environmental interactions each requires the production, assembly, operation, and regulation of many thousands of components, and they do so with remarkable fidelity in the face of many environmental cues and challenges (Zhu et al., 2007). Since the end of the 1990s, there has been a flood of interaction data for proteins, carbohydrates, DNA, RNA, lipids and other molecules (Yamada and Bork, 2009) Each completed genome sequencing project generates large data sets of different interactions, specially protein-protein interactions (PPI) (Yamada and Bork, 2009). Adding other types of interactions like metabolic networks, signalling networks, transcription regulatory networks the analysis problem of networks becomes critical due to their size. Still the network representation of mentioned interactions allows application of graph theory (Strogatz, 2001; Watts and Strogatz, 1998) and graph-topological analysis (Assenov et al., 2008) to the biological data to get insight into the global network structure.

Networks have "emergent" properties that are distinct from those of their individual components. Therefore networks have to be studied as systems. Emergent properties are non-linear, aggregated and combinatory effects generated by the interaction of the components of the network. For example, properties such as topology, information flow and the stable state of a network can only be detected at the network level, not by examining the individual components such as genes or proteins. The structural and dynamic features of genetic networks ultimately contribute to biological functions, robustness and evolvability of the networks (Han et al. 2004).The topological measures can capture the cellular features of cellular networks and provide broad insight into cellular evolution, molecular function, network stability, and dynamic responses (Chen et al., 2009).

The author in this paper reviews and classifies the most popular measures and properties of biochemical network topology and the most relevant freely available software tools for its analysis.

## 2. Measures and properties of network structure

Examining scientific literature, publications and analyzing software tools, author found many network structure measures and properties, which can be categorized in five groups: topological parameters, topological features, network metrics, network motifs and quantitative parameters of whole network structure that are the global topological parameters (see Fig.1). Some of measures and properties will be explained below.

### 2.1. Topological parameters

Topological parameters can be divided in two groups – local and global parameters (Durek and Walther, 2008), corresponding to the measurable element. Local topological parameters characterize individual network components while global parameters describe the whole network. One of local parameters is the **degree of a network node**. The degree (or connectivity) (Barabási and Oltvai, 2004; Robins et al., 2008; Yamada and Bork, 2009) of an undirected network node, $k_i$, is

the number of edges (links) that it has with other nodes (see Fig.2) that is incident with *i*:

$$k_i = \sum_j^n k_{ij} \quad (1)$$

For directed network degree is separated in two types: incoming (in-degree) and out coming (out-degree) degree, depending on the direction of interactions (Hu et al., 2005). A degree is also a feature that distinguishes hubs (highly connected nodes) from leaves or orphans (weakly or non-connected nodes) in the network (Zinovyev et al., 2008). In protein interaction and genetic interaction networks, for example, the degree of a hub (highly connected node) is often hub's importance and essentiality for cell function (Hu et al., 2005), process or whole system.

**Degree distribution** $d_k$ is the number of nodes with degree $k$ (k=1,2,…n) (Chen et al., 2009; Robins et al., 2008). For directed networks the degree distribution is separated into in-degree and out-degree distribution.

Let K be the degree of a network node. Then a statistical model for the degree distribution is represented by:

$$P(K = k) = f(k) = \frac{N_k}{N} \quad (2)$$

,where *f(k)* is a probability distribution
$N_k$ – a number of nodes with degree $k=1,2…n$
$N$ – the total number of nodes.

The distribution of degrees *f(k)* in undirected network, gives the probability that a selected node has degree *k* (Barabási and Oltvai, 2004). In the case of directed networks one needs to consider two distributions, $P(k_{in})$ and $P(k_{out})$ (Boccaletti et al., 2006).

The degree distribution of many types of real-life networks, such as metabolic or signalling, scientific collaboration networks is called a power law (Robins et al., 2008; Zhang and Shakhnovich, 2008):

$$P(K = k) \sim k^{-\gamma} \quad (3)$$

,where γ - a constant or the degree exponent.

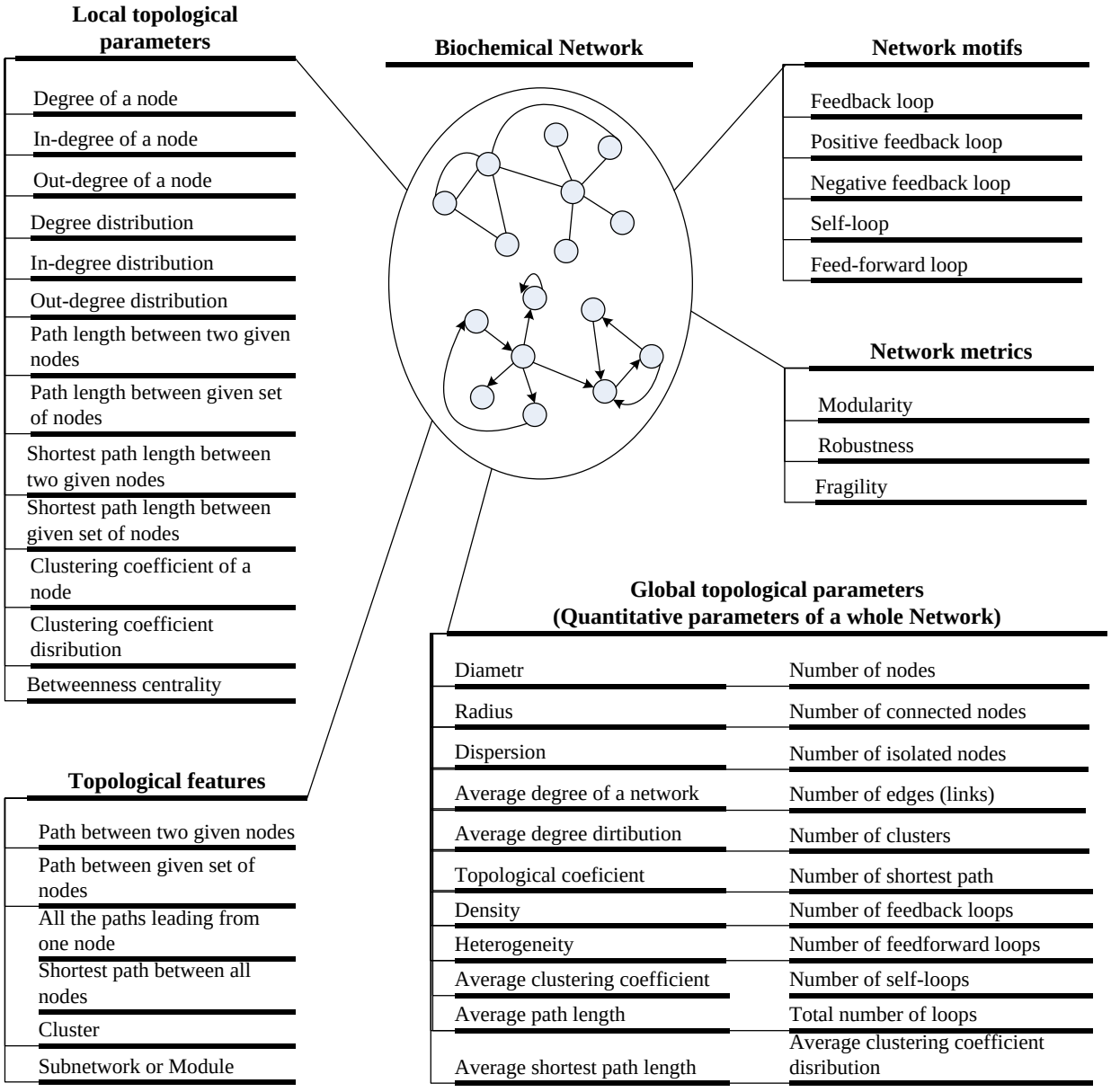


Fig.1. **Biochemical network measures and properties** (Rubina and Stalidzans, 2010a)

A power-law degree distribution indicates that a few hubs hold together numerous small components or nodes (Barabási and Oltvai, 2004). In scale-free networks most nodes have only one or two functional links, whereas a small number of nodes, the hubs, have many links (Han, 2008). Nearly all biological networks, including regulatory, interactome and metabolic networks are scale-free (Barabási and Oltvai, 2004; Boccaletti et al., 2006). Still there are other types of networks like random network and hierarchical network (Yamada and Bork, 2009).

## 2.2. Topological features

Studying the function of pathways, the property of interest is often how a given gene or protein is related to (or responds to) an up- or downstream signal. Given a large data set of interactions, it may be useful in some contexts to find the most direct path between two genes, proteins, complexes or pathways; for example, the overall lengths of such pathways may be related to the immediacy or breadth of signal response (Hu et al., 2005).

According to the graph theory **the path** (Wilson, 1972) is the sequence of nodes from $n_0$ to $n_k$. There can be different types of paths: chain (have all different edges), simple chain (have all different nodes), closed chain or cycle (starts and ends with the same node). Cycles compose the separated group of network measures – network motifs.

**The path between two given nodes** (see Fig.3). In case of signalling networks, the computation of all paths between pair of species helps to recognize all the different ways in which a signal can propagate between two nodes, e.g. all the different ways by which a certain transcription factor (or any other species from the output) can be activated or inhibited by signals riving the input layer.

The path length $l_{ij}$ is the number of edges (or links) in path from node *i* to *j*.

**The shortest path between two nodes** is the path between two nodes in a network with a smallest number of steps compared to alternative paths between the same nodes (Yamada and Bork, 2009).

The shortest path between given set of nodes is the path that connects all the nodes of given set with smallest number of steps.

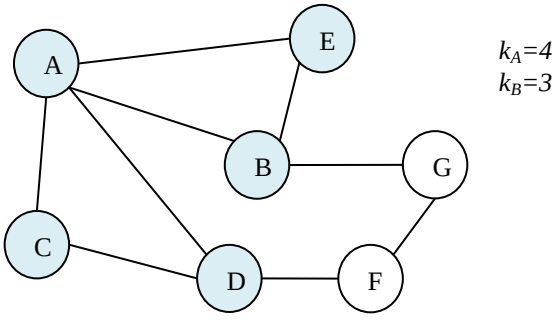


Fig.2. **Undirected network.**

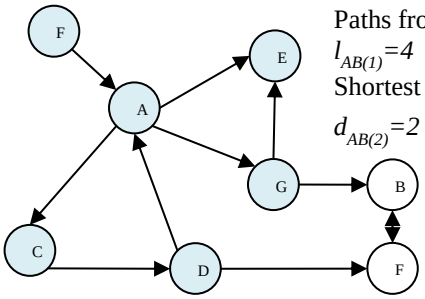


Fig.3. **Directed network**

## 2.3. Network motifs

Cellular networks are composed of complicated interconnections among nodes, and some subnetworks (in case of graph theory, subgraphs) of particular functioning are often identified as network motifs (Kim et al., 2008). **Network motifs** are the simple building blocks (Milo et al., 2002) of complex biochemical networks and are defined as patterns of interconnections that recur in many different parts of a network. The biochemical networks are composed of three highly significant motifs that repeatedly appear: feedback, feed-forward and self-loops. Each network motif has a specific function and play important dynamical roles in behaviour regulation of biological processes.

Biological systems are known to be considerably robust to environmental changes and genetic perturbations (Barabási and Oltvai, 2004; Kitano, 2004, 2007; Kwon and Cho, 2007a). Robustness is a fundamental feature of complex systems that allows them to maintain its functions despite external and internal perturbations (Kitano, 2004). The main mechanism that ensures the robustness of a system is a *system control* that consists of *negative* and *positive feedback.* Presence of feedback is the important party of control in biological systems (Rubina and Stalidzans, 2012). Negative feedback promotes restoration of an initial condition of system. Positive feedback withdraws system all further from an initial condition and strengthens the processes of ability to live.

From the viewpoint of network structure feedback is organized on *feedback loops*. According to the graph theory *feedback loop* is closed simple cycle (Barabási and Oltvai, 2004; Kwon and Cho, 2007a) of any length (Hallinan and Jackway, 2007) with the set of nodes where the nodes are not revisited except the starting and ending nodes. Exploring dynamic models of biochemical networks, researchers have established that feedback loops are very often found as a coupled structure in cellular circuits. *Coupled feedback loop* is closed cycle with the set of nodes where each node is visited twice (in reverse order) except one node in the middle of loop, for example, coupled feedback loop A→B→C→D→C→B→A. Coupled feedback loops can be positive and negative and can form three types of coupled structures (Kim et al., 2007; Kim et al., 2008): positive-positive, positive-negative and negative-negative structures (see Fig.4).

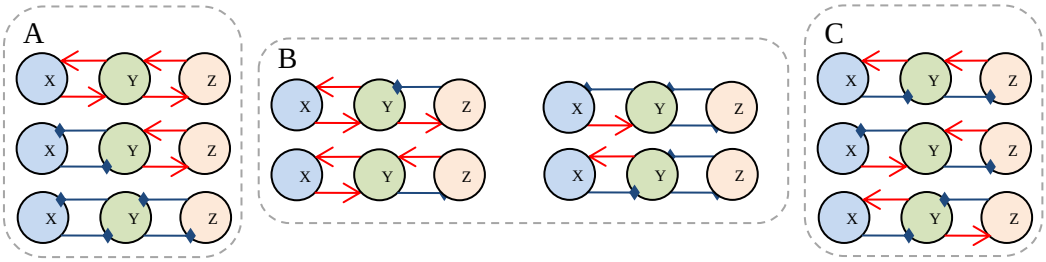


Fig.4. **Network motifs of coupled feedback loops** (Kim et al., 2008).
**(*A*) Positive-Positive structures. (*B*) Positive-Negative structures. (*C*) Negative-Negative structures.**

Kwon and colleagues (Kwon and Cho 2007) have verified hypothesis on the relationship between feedback loops and the robustness of a network by employing Boolean network models. They found that three distinct feedback loops are responsible for genetic regulation, mRNA attenuation, and enzyme inhibition that regulate tryptophan concentrations in Escherichia coli. The complex regulatory network formed by the feedback loops induces a rapid and stable response, while being robust against uncertainties (Kwon and Cho, 2007a; Venkatesh et al., 2004).

Kim with colleagues (Kim et al. 2007) suggests that coupled positive and negative feedback loops form essential signal transduction motifs in cellular signaling systems or signaling pathways. They performed mathematical simulations and investigations into various experimental evidences, and found that positive and negative coupled feedback circuits can rapidly turn on a reaction to a proper stimulus, robustly maintain its status, and immediately turn off the reaction when the stimulus disappears. In other words, coupled feedback loops enable cellular systems to produce perfect responses to noisy stimuli with respect to signal duration and amplitude (Kim et al. 2007). Likewise Shi with colleagues has proved (Shi et al., 2012) that coupled positive feedback loops can generate reversible and irreversible switch. And coupled positive feedback loops can strengthen bistable, enlarge signal and extend the signal reaction time. It means, that coupled positive feedback loops play an important role in regulation of biological behaviours. Therefore their recognition in biochemical networks is important task.

## 3. Tools for structure analysis of biochemical networks

According to the earlier performed analysis of existing tools (Rubina and Stalidzans 2010), the best tools for topology analysis is Cytoscape with plugins BiNoM and NetworkAnalyzer. Tools with good performance are also VisANT, Biological Networks and CelNetAnalyser. Comparative analysis of these tools demonstrates can simplify the choice of appropriate tool for solving of a particular task.

**CellNetAnalyser** is free for academic use package for MATLAB. *CNA* provides an environment for structural and functional analysis of biochemical networks such as metabolic, signalling and regulatory networks. It includes metabolic flux analysis, analysis of basic topological / structural properties, metabolic pathway analysis and for signal flow (signalling, regulatory) networks including analysis of interaction graphs, analysis of logical (boolean) interaction networks.

**Cytoscape** is an open source tool for visually exploring of biological networks, that support the SBML and BIOPAX standards. Cytoscape specializes in the representation of interaction networks and includes many powerful network display styles (Sudermann and Hallett, 2007). Automatic layout algorithms help to organize massive amounts of interaction data relating to a set of molecules (Chen et al, 2009). **NetworkAnalyzer** is the versatile Cytoscape plug-in (Assenov et al., 2008) that computes a comprehensive list of simple and complex topology parameters (single values and distributions) for directed and undirected networks using efficient graph algorithms. **BiNoM** is a Cytoscape plug-in, developed to analyze a structure of the networks.

**VisANT** is free and open source integrative web-based software platform for the visualization, mining, analysis and modelling of the biological networks. Visant allows to create multi-scale networks, represent many types of biological data, such as biomolecular interactions, cellular pathways and functional modules and provides a visual interface for combining and annotating network data, supporting function and annotation data for different genomes from the Gene Ontology and KEGG databases. It contains statistical and analytical tools needed for extracting topological properties of the user-defined networks.

**Biological networks** is free for academic use application for visualization and analysis of biological pathways. It is a graph-based system for creating a combined database of biological pathways, gene regulatory networks and protein interaction maps. After importing expression data, users can apply sorting, normalization and clustering algorithms on the data and then create various tables, heat maps and network views of the data.

Next tables provide summary of topological parameters (Table 1, Table 2) and features of network structure (Table 3) that can be analyzed by selected software tools, dividing topological parameters in two main groups – simple and complex parameters.

Table 1.

**Summary of computed simple topological parameters by software tools Visant, Cytoscape with BiNoM,CellNetAnalyzer and Biological Networks.**

| | Local topological parameters | | | | Global topological parameters | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Parameters / Tools | Degree of a node | In-degree of a node | Out-degree of a node | Clustering coefficient of node | Average number of neighbours | Network diameter | Network radius | Density | Centralization | Heterogeneity | Number of nodes | Number of edges | Number of self-loops | Number of connected nodes | Number of isolated nodes | Number of shortest paths |
| **Visant** | ● | ● | ● | | | | | | | | | | | | | |
| **Cytoscape with BINOM & Network-Analyser** | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | ● | | ● |
| **CellNet-Analyser** | | | | | | ● | | | | | ● | | | | | |
| **Biological Networks** | | | | ● | | | | | | | | | | | | |

Table 2.

**Summary of computed complex topological parameters by software tools Visant, Cytoscape with BiNoM,CellNetAnalyzer and Biological Networks.**

| | Local topological parameters | | | | | | Global topological parameters | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Parameters / Tools | Node degree distribution | Node in-degree distribution | Node out-degree distribution | Clustering coefficient distribution | Shortest path lengths | Shortest path distribution | Average degree distribution | Neighborhood connectivity's | Neighbours connectivity | Average clustering coefficient | Average clustering coefficient distribution | Topological coefficients | Average path length | Average shortest path length | Shared neighbours of two nodes |
| **Visant** | ● | ● | ● | ● | | | ● | | | ● | ● | | | | |
| **Cytoscape with BINOM & Network-Analyser** | ● | ● | ● | | ● | ● | | ● | ● | | | ● | | ● | ● |
| **CellNet-Analyser** | | | | | ● | | | | | | | | ● | | |
| **Biological Networks** | ● | | | | ● | | | | | | | | | | |

Table 3.

**Summary of computed network motifs and topological features by software tools Visant, Cytoscape with BiNoM,CellNetAnalyzer and Biological Networks**

| | Topological features | | | | | | | | | | Network motifs | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Parameters / Tools | Shortest path between all nodes | Shortest path between two given nodes | Shortest path between given set of nodes | Optimal and sub-optimal shortest paths | All the paths leading from one node | All non-intersecting paths | Path finding between two given nodes | Path finding between given set of nodes | Cluster finding | Cycle clustering and decomposition | Feedback loops finding | Feed-forward loops finding | Self-loops finding |
| **Visant** | ● | ● | ● | | | | ● | ● | | | ● | ● | ● |
| **Cytoscape with BINOM & Network-Analyser** | ● | ● | | ● | ● | ● | | | ● | ● | | ● | |
| **CellNet-Analyser** | ● | ● | | | | | | ● | | | ● | ● | |
| **Biological Networks** | ● | ● | | | | | | ● | ● | | | ● | |

## 4. Conclusion

The topology of biochemical networks can be analysed using tens of measures and parameters. Analysis of software tools Visant, Cytoscape with BINOM and NetworkAnalyser, CellNetAnalyser and Biological Networks gives detailed overview about the functionality of software tools as well as their specialisation on determination of topological measures and parameters. Generally it is concluded that all the mentioned software tools can be involved in analysis of at least some measures and parameters of all five groups.

Software tools Visant and Cytoscape with BINOM and NetworkAnalyser plug-ins has the highest number of calculated parameters among the other compared software tools with relatively high number of simple topological parameters.

Software tools CellNetAanalyser and Biological Networks relatively more concentrate on calculation of complex topological parameters, network motifs and topological features.

All the mentioned software tools can calculate shortest path between all nodes, shortest path between two given nodes and perform finding of feed-forward loops.

## Acknowledgments

This work and academic study is funded by a project "Support for doctoral studies in LUA"/ 2009/0180/1DP/1.1.2.1.2/09/IPIA/VIAA/017" agreement Nr. 04.4-08/EF2.D1.D6".

## References

Assenov, Y., Ramírez, F., Schelhorn, S.-E., Lengauer, T. and Albrecht, M. (2008), "Computing topological parameters of biological networks" *Bioinformatics*, Vol. 24 No. 2, pp. 282–284. doi:10.1093/bioinformatics/btm554

Barabási, A.-L. and Oltvai, Z.N. (2004), "Network biology: understanding the cell's functional organization" *Nature reviews. Genetics*, Vol. 5 No. 2, pp. 101–13. doi:10.1038/nrg1272

Boccaletti, S., Latora, V., Moreno, Y., Chavez, M. and Hwang, D. (2006), "Complex networks: Structure and dynamics" *Physics Reports*, Vol. 424 No. 4-5, pp. 175–308. doi:10.1016/j.physrep.2005.10.009

Chen, L., Wang, R.-S. and Zhang, X.-S. (2009), "Biomolecular networks: Methods and Applications in Systems Biology" *Jersey, John Wikey & Sons*; Inc.; Hoboken; New.

Durek, P. and Walther, D. (2008), "The integrated analysis of metabolic and protein interaction networks reveals novel molecular organizing principles" *BMC Systems biology*, Vol. 2, p. 100. doi:10.1186/1752-0509-2-100

Gehlenborg, N., O'Donoghue, S.I., Baliga, N.S., Goesmann, A., Hibbs, M., Kitano, H., Kohlbacher, O., et al. (2010), "Visualization of omics data for systems biology" *Nature methods*, Nature Publishing Group, Vol. 7 No. 3, pp. S56–68. doi:10.1038/nmeth.1436

Hallinan, J.S. and Jackway, P.T. (2007), "Network Motifs, Feedback Loops and the Dynamics of Genetic Regulatory Networks" *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pp. 1–7.

Han, J.-D. (2008), "Understanding biological functions through molecular networks" *Cell research*, Vol. 18 No. 2, pp. 224–37. doi:10.1038/cr.2008.16

Han, J.D., Bertin, N., Hao, T., Goldberg, D.S., Berriz, G.F., Zhang, L.V., Dupuy, D., et al. (2004), "Evidence for dynamically organized modularity in the yeast protein-protein interaction network" *Nature*, Vol. Jul 15, 43, pp. 88–93.

Hu, Z., Mellor, J., Wu, J., Yamada, T., Holloway, D. and Delisi, C. (2005), "VisANT: data-integrating visual framework for biological networks and modules" *Nucleic acids research*, Vol. 33 Web Server issue, pp. W352–357. doi:10.1093/nar/gki431

Ideker, T., Galitski, T. and Hood, L. (2001), "A new approach to decoding life: systems biology" *Annual review of genomics and human genetics*, Vol. 2, pp. 343–372. doi:10.1146/annurev.genom.2.1.343

Kim, D., Kwon, Y.-K. and Cho, K.-H. (2007), "Coupled positive and negative feedback circuits form an essential building block of cellular signalling pathways" *BioEssays*: news and reviews in molecular, cellular and developmental biology, Vol. 29 No. 1, pp. 85–90. doi:10.1002/bies.20511

Kim, J.R., Yoon, Y. and Cho, K.H. (2008), "Coupled feedback loops form dynamic motifs of cellular networks" *Biophysical journal*, Vol. 94 No. 2, pp. 359–365. doi:10.1529/biophysj.107.105106

Kitano, H. (2004), "Biological robustness" *Nature reviews.* Genetics, Vol. 5 No. 11, pp. 826–837. doi:10.1038/nrg1471

Kitano, H. (2007), "The theory of biological robustness and its implication in cancer" *Nature Rewievs*. Genetics, Vol. 5 No. 11, pp. 826–837.

Kwon, Y.-K. and Cho, K.-H. (2007), "Analysis of feedback loops and robustness in network evolution based on Boolean models" *BMC bioinformatics*, Vol. 8, p. 430. doi:10.1186/1471-2105-8-430

Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D. and Alon, U. (2002), "Network motifs: simple building blocks of complex networks," *Science* (New York, N.Y.), Vol. 298 No. 5594, pp. 824–827. doi:10.1126/science.298.5594.824

Robins, G., Pattison, P. and Koskinen, J. (2008), "Network degree distributions" *Technical report*, pp. 1–9.

Rubina, T. and Stalidzans, E. (2010a), "Tological features and parameters of biochemical network structure" EUROSIS, *Proceedings of Industrial Simulation Conference*, Budapest, Hungary, pp. 228–236.

Rubina, T. and Stalidzans, E. (2010b), "Software tools for structure analysis of biochemical networks" *Proceedings of Applied Information and Communication Technologies (AICT)*, Jelgava, Latvia, pp. 33–49.

Rubina, T. and Stalidzans, E. (2012), "Evolution of alternative control loops of biological systems" *Proceedings of Applied Information and Communication Technologies (AICT)*, Jelgava, Latvia, pp. 317–324.

Shi, F., Zhou, P. and Wang, R. (2012), "Coupled positive feedback loops regulate the biological behavior" *IEEE 6th International Conference on Systems Biology (ISB)*, Ieee, pp. 169–173. doi:10.1109/ISB.2012.6314131

Strogatz, S.H. (2001), "Exploring complex networks" *Nature*, Vol. 410 No. 6825, pp. 268–76. doi:10.1038/35065725

Suderman M, Hallett M. (2007), "Tools for visually exploring biological networks" Bioinformatics, Vol. 23 No 20, pp.2651-2659. doi:10.1093/bioinformatics/btm401

Venkatesh, K.V., Bhartiya, S. and Ruhela, A. (2004), "Multiple feedback loops are key to a robust dynamic performance of tryptophan regulation in Escherichia coli" *FEBS letters*, Vol. 563 No. 1-3, pp. 234–240. doi:10.1016/S0014-5793(04)00310-2

Watts, D.J. and Strogatz, S.H. (1998), "Collective dynamics of 'small-world' networks" *Nature*, Vol. 393, pp. 440–442.

Wilson, R.J. (1972), Introduction to graph theory, *New York*, Academic Press.

Yamada, T. and Bork, P. (2009), "Evolution of biomolecular networks: lessons from metabolic and protein interactions" *Nature reviews*. Molecular cell biology, Vol. 10 No. 11, pp. 791–803. doi:10.1038/nrm2787

Zhang, J. and Shakhnovich, E.I. (2008), "Sensitivity-dependent model of protein–protein interaction networks" *Physical Biology*, Vol. 5. doi:10.1088/1478-3975/5/3/036011

Zhu, X., Gerstein, M. and Snyder, M. (2007), "Getting connected: analysis and principles of biological networks" *Genes & development*, Vol. 21 No. 9, pp. 1010–1024. doi:10.1101/gad.1528707

Zinovyev, A., Viara, E., Calzone, L. and Barillot, E. (2008), "BiNoM: a Cytoscape plugin for manipulating and analyzing biological networks" *Bioinformatics*, Vol. 24 No. 6, pp. 876–877. doi:10.1093/bioinformatics/btm553