CORUNNER: MULTIPLE OPTIMIZATION RUN MANAGER FOR COPASI SOFTWARE

JURIJS SULINS, EGILS STALIDZANS

Biosystems Group, Departnment of Computer Systems, Latvia University of Agriculture, Latvia
SIA TIBIT, Latvia
info@tibit.lv

Abstract: *Numerical optimization based design tasks of dynamic models of biochemical networks mainly are solved using global stochastic optimization methods which can deliver relatively quick improvement of the objective function in short time but it cannot guarantee finding of optimal solution. In case of large models or complicated dynamics the question about appropriate duration of optimization runs become important question in case of necessity to analyze high number of combinations of adjustable parameters. This task becomes even more complex if the optimal solution has to fulfil the steady state condition which is typical in case of biotechnological applications of design tasks as steady state precondition rejects many combinations of adjustable parameter values thus making the optimization slower.*
Duration of one optimization run therefore can take time up to several days or weeks for one combination of adjustable parameters which make the procedure inefficient. Application of parallel optimization runs is proposed to reduce the stochastic impact of a single optimization run.
A software CoRunner is developed to apply parallel runs automatically collaborating with software COPASI for automation of optimization procedure. In case of necessity the dynamics of parallel optimization runs, which are stored in form of COPASI reports, can be visualized with software ConvAn for detailed analysis of optimization performance. CoRunner automatically stops the optimization in case of consensus of all the parallel optimization runs and returns COPASI model with the set of best values of adjustable parameters found.

Keywords: biochemical networks, stochastic numerical optimization, parameter estimation, design task, convergence dynamics.

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. Important class of tasks is the optimization of the model in the allowed space of adjustable parameters is the optimization that can be used both in parameter estimation and design tasks (Moles et al., 2003).
Dynamic models give detailed quantitative 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 representing 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 generally and in optimization tasks in particular.
Mostly the numerical optimization methods are used in optimization tasks of biochemical networks. They can be divided in 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 minimum. There are two classes of global numerical optimization methods: deterministic ones and the stochastic ones. The advantage of some of deterministic methods is the guaranteed reach of global optima 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 optima close value (Banga, 2008; Moles et al., 2003).
The convergence of global stochastic optimization methods is analyzed in case of parameter estimation tasks (Baker et al., 2010; Balsa-Canto et al., 2010; Balsa-Canto et al., 2008; Mendes and Kell, 1998; Moles et al., 2003) while less research is found on convergence in case of design 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). Condor COPASI project can be mentioned as a case where parallel optimization runs are applied using web interface (http://code.google.com/p/condor-copasi/). Variable convergence speed and stagnation at non optimal values are mentioned as critical in design problems of biochemical networks (Mozga and Stalidzans, 2011b; Mozga and Stalidzans, 2011c) where even relatively small number (5-15) of adjustable parameters of the model cause hundreds or thousands of combinations to be explored. High variability of convergence speed is reported also by Nikolaev: optimization runs can last from minutes up to 90 hours (Nikolaev, 2010). Software ConvAn (Kostromins et al., 2012) is proposed for statistical analysis of convergence dynamics is proposed to analyze and compare different settings of optimization tasks. The combinatorial explosion of adjustable parameter sets force to look for efficient approaches to reduce

necessary time either by rejecting some combinations of adjustable parameters or by reducing the optimization time.

Therefore an automatic termination of optimization run would be of advantage to spend exactly the necessary time for particular model, criteria, set of parameters and optimization method. We propose to use a desktop application software that collaborates with COPASI command line version (Hoops et al., 2006) initiating parallel optimization runs and terminates the optimization automatically when a termination criteria is satisfied.

As set of criteria is proposed to stop a set of optimization runs when it is assessed that no major improvements are expected. The first criteria 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; Mozga and Stalidzans, 2011c; Mozga et al., 2011). The second criterion is the end of pre-set maximal duration of optimization runs in case if no consensus is reached.

Use of these criteria's for automatic termination of optimization runs reduce the length of optimization experiment by more intensive use of computational resources due to parallel optimization runs.

Software description

Software development and compatibility

CoRunner works on machines running Windows XP, Vista or 7 and Microsoft .NET Framework version 3.5 (available on http://www.microsoft.com/net). To avoid unexpected errors set point as your decimal separator that means that user must check under Regional Setting in Windows OS if decimal separator is point.

CoRunner installation package contains Copasi SE (command line version of Copasi) 35. Build version optimization tool files.

Collaboration with COPASI and ConvAn

CoRunner needs the Copasi file for execution. Copasi file must have fully configured all parameters for optimization task and checked check box for executable in optimization task must be checked. Software works fine for both of optimizations types (minimization or maximization). Please give the most attention to optimization report configuration: separator must be "tab"; in body tab first must be the time value and second the criteria (for example best value); software will ignore header configuration but it's recommended to leave this section empty; also strongly recommended to leave footer configuration empty otherwise it may cause an error. CoRunner will generate the name for report file automatically, so it's not necessarily to set name of report file.

When all optimizations are done Copasi generated report files are suitable to perform convergence analyse using ConvAn software (Kostromins et al., 2012).

Functionality of CoRunner

First of all it's necessary to fetch the Copasi file (*.cps), to do so use yellow folder button and select file you need. When it's done in text box next "Path to Copasi file:" label will appear path to your selected Copasi file.

That's enough to run program with default settings. Press the "Go!" button and the program will start 5 parallel runs and criteria checking. All report files will be stored in "tmp" sub directory of CoRunner installation directory. While program is running on form will appear inscription: "Program is RUNNING!"

Program cycles label indicates how many minutes program is running.

CoRunner's parameters configuration is described below.

First is "Number of parallel optimization runs" this value means how many runs and report files will be created and executed. It's suggested to set this value not more than one less as total count of CPU core on machine.

"Waiting for first increase of criteria value" – time in minutes that given to all parallel optimizations to reach at least one first criteria change. This option is significant for models that have a risk don't reach any possible criteria value.

"Consensus criteria" this configuration indicates maximum offset of best criteria (in percentage).

"Consensus delay time" when criteria consensus is reached and not changed during this amount of time all optimizations will be stopped.

"Maximal duration of optimization" is the maximal time for all optimizations to reach the consensus. If consensus didn't reach during this amount of time CoRunner will consider that optimizations aren't perspective or there is a stagnation and stop all of optimizations automatically in preselected time.

"Keep new Copasi files" if this check box is checked then after stopping all of optimizations CoRunner will create new Copasi files for all optimizations. This option is suggested when "update model" is checked in source Copasi file.

Last configuration parameter is check box - "Insert last criteria value before stopping". If this value is checked then before stopping optimization best value for each optimization will be inserted in report file at stopping time. This option is suggested if further report files will be used in ConvAn for analyses.

CoRunner software and manual available at Tibit home page (http://tibit.lv/corunner/)

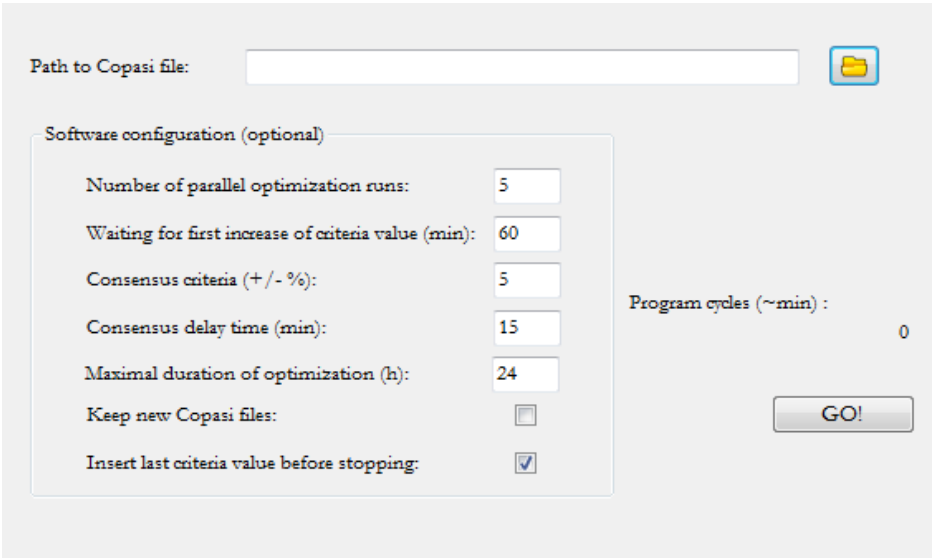

Fig. 1. **User interface of CoRunner**

Applications

Several cases of results of CoRunner applications are demonstrated. The figures are generated by the software ConvAn using data from report files created by CoRunner.

Case of good performance of optimization can be seen in the Figure 2 and Figure 3. In both cases the optimization is stopped when all the runs reach consensus (best values are within 3% area). It can be seen that the number of parallel runs (5 runs in Figure 2 and 10 runs in Figure 3.) in this particular case does not influence the descriptive statistics of parallel runs. Still the optimization length in case of ten reactions (Fig.3) was almost two times longer.

Stagnation of optimization method (Fig.4) is very important case as optimization results have to be interpreted very carefully (Mozga and Stalidzans, 2011a; Mozga and Stalidzans, 2011b; I Mozga et al., 2011) as stagnation indicate that particular optimization method is not well suited to the particular model or optimization task. Therefore even the best value of the parallel optimization runs may not be the best possible value of objective function (I Mozga and Stalidzans, 2011c).

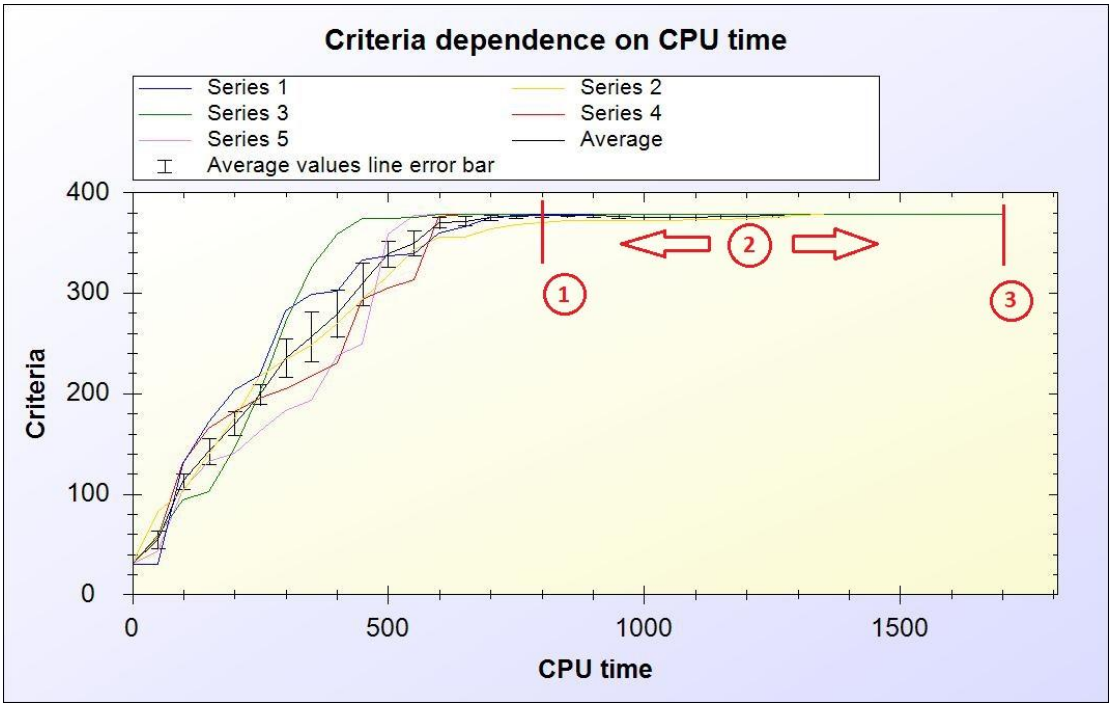

Fig. 2. **Consensus case when all the five optimization runs have reached the consensus range**

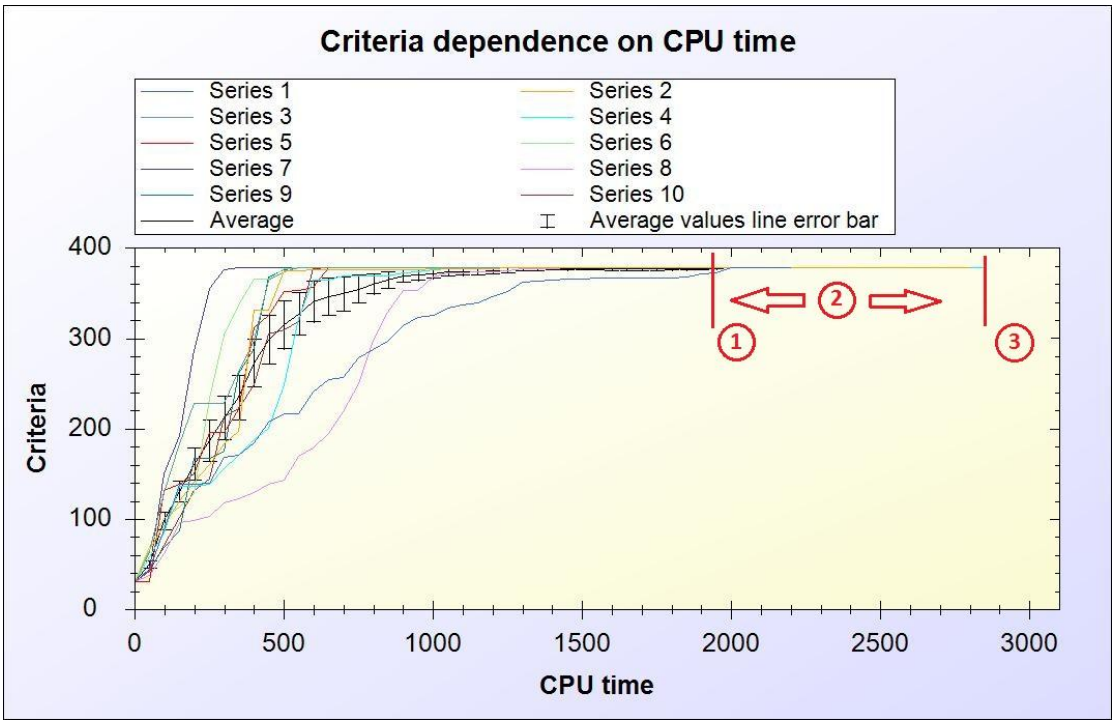

Fig. 3. **Consensus case when all the ten optimization runs have reached the consensus range**

Consensus case when all the five (Fig. 2) and all the ten (Fig. 3) optimization runs have reached the consensus range (1 – best values of all runs lie within 3% consensus range; 2 – preset consensus delay time (15min or 900s); 3 – all optimization auto stop after reaching consensus and passing the consensus delay time). Error bars represent standard deviation.

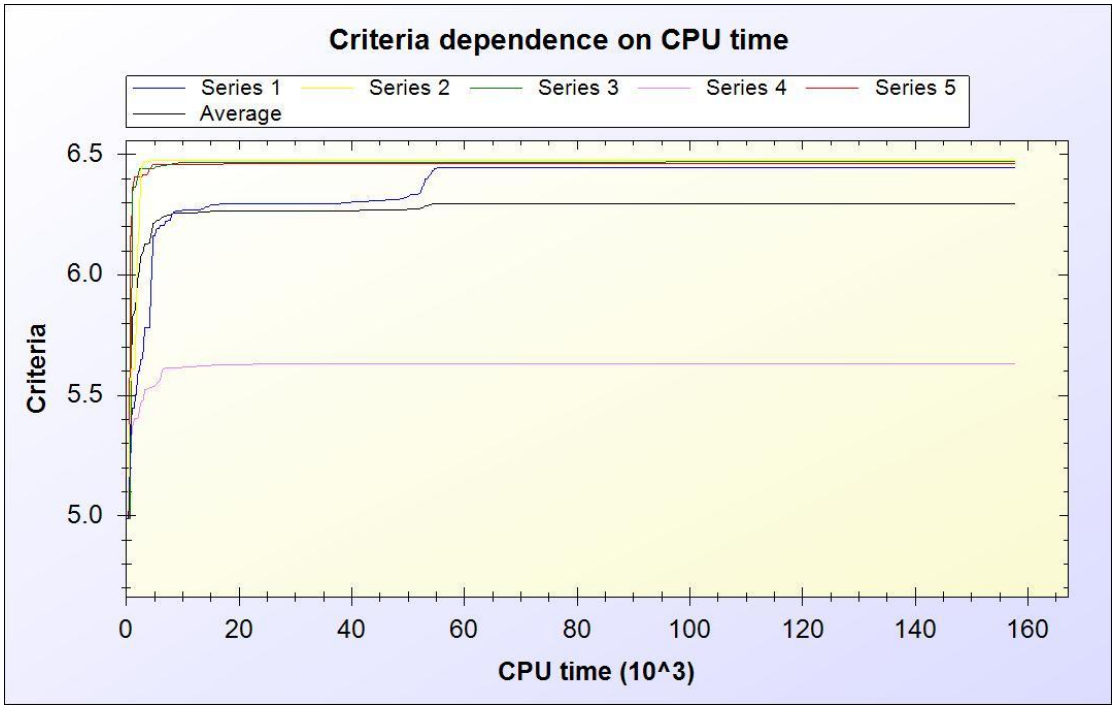

Fig. 4. **Stagnation example when different optimization runs stagnate at different values long time**

Conclusion

Optimization based design tasks of dynamic models of biochemical networks mainly are solved using global stochastic optimization methods which can deliver relatively quick improvement of the objective function in short time but it cannot guarantee finding of optimal solution.

Application of number of parallel optimization runs is proposed to shorten the optimization time and detect failure of optimization method. A software CoRunner is developed to apply this method automatically

collaborating with software COPASI for automation of optimization procedure. In case of necessity the dynamics of parallel optimization runs can be visualized with software ConvAn for detailed analysis of optimization performance.

Suitability of optimization method for particular model and optimization task setting can be confirmed in case of convergence of all optimization runs to a consensus value. Stagnation of at least one optimization run indicates that method is not well suited for particular task.

References

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*, *7*(3), 1-9. doi: 10.2390/biecoll-jib-2010-133.

Balsa-Canto, E., Alonso, A. a, & Banga, J. R., 2010. An iterative identification procedure for dynamic modeling of biochemical networks. *BMC systems biology*, *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*, *2*, 26. doi: 10.1186/1752-0509-2-26.

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

Hoops, S., Sahle, S., Gauges, R., Lee, C., Pahle, J., Simus, N., et al., 2006. COPASI--a COmplex PAthway SImulator. *Bioinformatics (Oxford, England)*, *22*(24), 3067-74. doi: 10.1093/bioinformatics/btl485.

Jamshidi, N., & Palsson, B. Ø., 2008. Formulating genome-scale kinetic models in the post-genome era. *Molecular systems biology*, *4*(171), 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*, *108*(1-3), 73-77. doi:10.1016/j.biosystems.2011.12.004

Mendes, P., & Kell, D., 1998. Non-linear optimization of biochemical pathways: applications to metabolic engineering and parameter estimation. *Bioinformatics (Oxford, England)*, *14*(10), 869-83. Retrieved March 27, 2011, from http://www.ncbi.nlm.nih.gov/pubmed/9927716.

Moles, C. G., Mendes, P., & Banga, J. R., 2003. Parameter estimation in biochemical pathways: a comparison of global optimization methods. (C. Skjoldebremd & G. Trystrom, Eds.)*Genome Research*, *13*(11), 2467-2474. Goetheborg, Sweden. doi: 10.1101/gr.1262503.

Mozga, I, & Stalidzans, E., 2011a. Optimization protocol of biochemical networks for effective collaboration between industry representatives, biologists and modellers. *9-th Annual Industrial Simulation Conference* (pp. 91-96). Venice, Italy: EUROSIS.

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

Mozga, I, & Stalidzans, E., 2011c. Convergence Dynamics of Biochemical Models To The Global Optimum. *3rd International Conference on E-Health and Bioengineering - EHB 2011* (pp. 227-230). IEEE.

Mozga, I, Kostromins, A., & Stalidzans, E., 2011. Forecast of Numerical Optimization Progress of Biochemical Networks. *Engineering for Rural Development* (pp. 103-108). Jelgava.

Nikolaev, E. V., 2010. The elucidation of metabolic pathways and their improvements using stable optimization of large-scale kinetic models of cellular systems. *Metabolic engineering*, *12*(1), 26-38. Elsevier. doi:10.1016/j.ymben.2009.08.010

Stelling, J., 2004. Mathematical models in microbial systems biology. *Current opinion in microbiology*, *7*(5), 513-8. doi: 10.1016/j.mib.2004.08.004.