Beilstein-Institut

ESCEC, March 19th – 23rd, 2006, Rüdesheim/Rhein, Germany

JWS Online: A Web-Based Tool for Curation, Review, Storage and Analysis of Kinetic Models

Jacky L. Snoep[1,2,3], Carel van Gend[1], Cor Stoof[3], Brett G. Olivier[1], Riaan Conradie[1], Franco B. Du Preez[1], Du Toit W.P. Schabort[1], Gerald Penkler[1] and Kora Holm[1]

[1]Triple-J group for Molecular Cell Physiology, Department of Biochemistry, Stellenbosch University, Matieland 7602, South Africa

[2]Manchester Centre for Integrative Systems Biology, Manchester Interdisciplinary Biocentre, 131 Princess Street, Manchester M1 7ND, U. K.

[3]Molecular Cell Physiology, Vrije Universiteit, De Boelelaan 1087, 1081 HV Amsterdam, The Netherlands

E-Mail: jls@sun.ac.za

Received: 9th November 2006 / Published: 31st August 2007

Abstract

In this contribution we report on the JWS Online project and the progress that has been made since the first ESCEC meeting. Whilst maintaining the same user interface, we have completely redesigned the server part of JWS Online, now a) using webMathematica as the interface between the HTML pages and the Mathematica [1] Kernel and b) storing all models as Mathematica packages, and c) using a PostgresQL [2] database to store a full description of each model.

In the last few years a number of new initiatives have started, of which some fulfil comparable roles to JWS Online and with some of which we collaborate. Here we compare JWS Online to these initiatives focusing on the three aims of JWS Online: 1) to be a repository for curated kinetic models of biological systems, 2) to be an easy to use simulator that can be accessed over the internet, 3) to help in the reviewing of manuscripts containing kinetic models.

Introduction

Mathematical Biology has a long history, especially in the field of population dynamics, with famous examples such as the description of Fibonacci for the growth of an idealized rabbit population and the Lotka–Voltera equations for predator–prey interactions. In these earlier models equations were often selected more on the basis of ease of use in mathematical analysis and less on knowledge of the biological system. In the neurosciences the work of Hodgkin and Huxley was a major breakthrough, not only for the understanding of the generation of the action potential but also in their approach to build a kinetic model of the neuron using kinetic parameters that were experimentally determined. More recently there has been a tremendous increase in the interest of applying kinetic models in the field of molecular and cellular biology. Whereas pioneering work in this field was done in the 1960 s by Chance, Garfinkel, Higgins and Hess e. g. [3], an enormous increase in the last decade in the construction of detailed kinetic models can probably be related to 1) the development in experimental fields (e. g. genomics) leading to detailed and information-rich data sets, 2) the increase in computing power and strength of simulation tools and 3) further development of strong analysis frameworks, (e. g. dynamical systems analysis, metabolic and hierarchical control analysis). The combined use of such theoretical, computational and experimental approaches has been characteristic for the field of **Systems Biology**, developed over the last five years, aiming at an understanding at a systemic level via the integration of our knowledge of the system's components (including their interactions).

Detailed kinetic models form a core component of **Systems Biology** studies. These models contain the experimental information on the components of the systems and their integration should result in the systemic **behaviour** observed for the complete system. These models are different from the traditional models that are made as simple as possible; such so-called core models are used to test a hypothesis or to illustrate a theoretical concept. Their simplicity makes core models amenable to robust mathematical analysis, without being sidetracked by unnecessary detail. Examples are the two and three variable models often used in bifurcation analysis of dynamic systems. Core models are important to get an understanding of the general behaviour of a system or a set of equations but it is often not possible to directly relate the model to experimental data and model validation is often made in more qualitative terms. In addition to these core models, systems biology has a need for a different kind of model, with a high level of detail and a direct, mechanistic interpretation of the model components. In **the Silicon Cell initiative** we have advocated the use of models with a high level of detail, containing experimentally determined parameter values (e. g. [4, 5]). We suggest measuring the model parameters of the isolated components (either *in vitro* or *in vivo*) and validating the model against the behaviour of the complete system, thus clearly separating model construction from model validation. In addition we suggested a modular approach, i. e. building detailed models of parts of the system, subsequently validating these models and combining them, followed by an additional round of validation. Ultimately such an approach would lead to a kinetic description of a complete system, for instance making a detailed kinetic model for the yeast *Saccharomyces cerevisiae*.

The Silicon Cell initiative would result in a significant number of models to be constructed. Even a simple unicellular system contains several thousand reactions, and a sensible split over modules would need to be made. Adding to the complexity is the capacity of living organisms to adapt themselves via regulation at the level of gene expression potentially any of the reaction steps can be modulated. This variable gene expression is one of the reasons one should model the cell in detail at the level of the enzyme catalysed reaction step. With time, a large collection of models, including metabolic, signal transduction, cell cycle and gene expression regulation models will be constructed which upon grouping will ultimately cover the complete cell.

To be able to link kinetic models together they must obey certain standards in terms of **annotation** (e. g. variable names should be identical) and the models should be described in a standard format (e. g. SBML [6]). In addition to the standardization of formats the models should also be available in **curated** form in repositories. In this contribution we highlight one initiative, JWS Online, which in addition to being such a repository for curated models is also a simulator with a web interface, making it possible to run the models in a browser. A third important aspect of JWS Online is its collaboration with scientific journals to assist in reviewing manuscripts that contain models. We start by describing JWS Online and its current set-up, focusing on the server side (the user interface was described in the last ESCEC contribution and has largely remained the same). Subsequently we will compare JWS Online with other initiatives that have comparable functionalities, i. e. the Virtual Cell, BioModels, DOQCS, Sigpath, JSim, ModelDB, Web-Cell and the SBML and CellMl repositories. We limit ourselves to these web-based initiatives, (and apologize for potential omissions), and have not included stand-alone simulators.

JWS Online

JWS Online [7] is hosted at the National Bioinformatics Node of the University of Stellenbosch. The first version of the web site went online in 2000 and since then a number of important updates have been made but the three main aims have remained the same. JWS Online is: 1) a repository of curated models, 2) a web-driven simulator and 3) a review facility for scientific journals. Models have been added steadily and currently 70 models are available, in three categories, Silicon Cell models, Core models and Demonstration models. JWS Online is mirrored at the Vrije Universiteit in Amsterdam (http://jjj.bio.vu.nl), at the Virginia Bioinformatics Institute (http://jjj.vbi.vt.edu) and at Manchester University (http://jjj.mib.man.ac.uk).

JWS Online works together with four journals to facilitate reviewing of manuscripts that contain kinetic models: *FEBS J.*, *Microbiology*, *IEE Proceedings Systems Biology* and *Metabolomics*. Authors who submit a manuscript containing a kinetic model are requested to submit their model to JWS Online (jls@sun.ac.za) in electronic format (i. e. SBML or JWS Online input form). Subsequently the model is converted into a Mathematica package that is stored in the JWS Online database. Using the JWS Online facility the simulations of the authors are repeated and if the results cannot be reproduced the authors are contacted to

resolve the problem. Once the model is curated in this way, a letter is sent to the reviewers stating how they can access the model on a secure site and reproduce the results of the authors and otherwise interrogate the model. Once the reviewers have come to a decision regarding the manuscript, the model is either moved to the public database or deleted.

JWS Online collaborates with a number of other initiatives: **the Silicon Cell** initiative, **Biomodels** (see below), **YSBN**, the Yeast Systems Biology Network (http://www.gmm.gu.se/YSBN/), and **HepatoSys**, the BMBF funded German systems biology competence network of hepatocytes (http://www.systembiologie.de/en/index.html), and the **COPASI** team (http://www.copasi.org/).

The functionality of JWS Online was discussed in the first ESCEC proceedings [8] and here we will only briefly summarize the functionality of the simulator and describe the way the server side of JWS Online works.

JWS Online set-up

The JWS simulation system is based on a client–server architecture, where commands issued by the client (a Java applet in a web browser) are fulfilled by an instance of Mathematica running on the server, see Fig. 1 for a flow diagram. This is facilitated by a webserver (Apache Tomcat [9]) running webMathematica, which is responsible for allocation of Mathematica kernels from a pool, accepting client commands and sending these to the Kernel for evaluation, and returning the results to the client.

The JWS models are stored as Mathematica packages. These include values for the model input parameters, and also define the functionality available for the model. In particular, functions may be defined which calculate and plot a time course of the model, display the steady state of the model, or display the results of a metabolic control analysis. The details of each of these calculations are specific to a particular model, and are described in the package. In addition, the package may define only a subset of these functions, depending on what is appropriate for the model. An SQL database contains a full text description of each model, as well as links to the Mathematica package for that model.

On visiting the JWS site, the user is presented with a welcome screen, displaying basic site information. The user may then opt to choose a model from the database. An initial selection page is displayed, in which the user may select any or all of model organism, model category and subcategory and model author.

The selection request is then sent to the web server, where a Python [10] script extracts from the database those models that satisfy the selection criteria. These are displayed in the user's browser as a list, from which the user may choose to display detailed information about a particular model, or opt to run a model.

The request to run a model is returned to the server, where the webMathematica kernel manager allocates a Mathematica kernel from the kernel pool. This kernel then loads the model from the Mathematica model description, and passes the model parameters to the

JWS Java applet, which is downloaded to the browser. The applet is configured according to the functionality available for the specific model chosen; certain models, for example, allow a time-course simulation, steady-state analysis and the determination of metabolic control analysis information, while others allow only a subset of these.

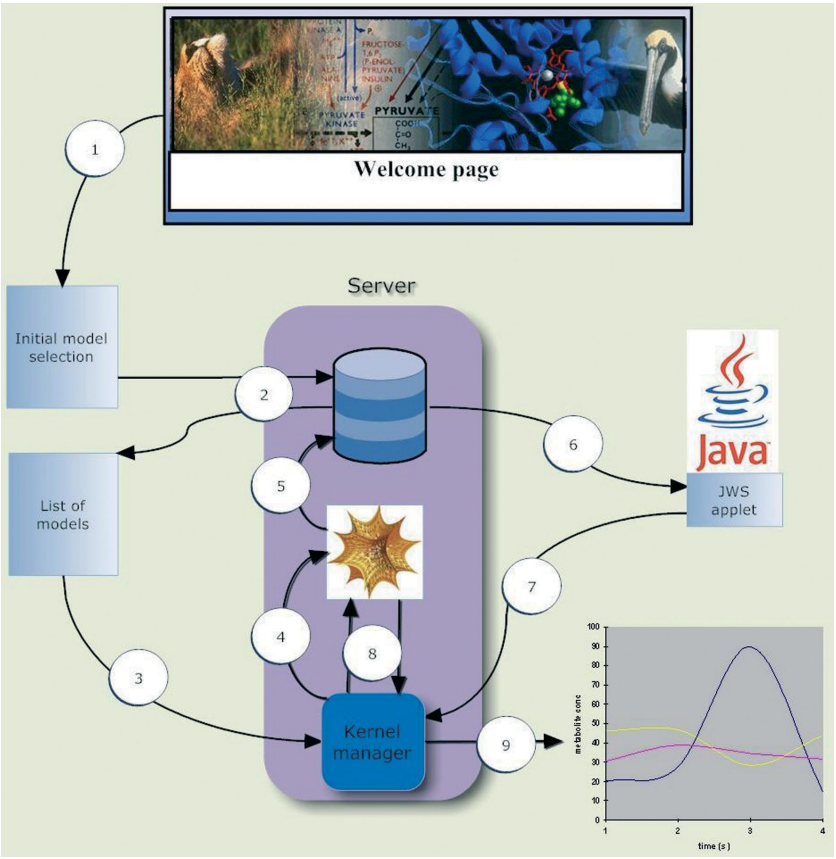

Figure 1. Flow diagram of the JWS Online set-up.

1) The user selects the link to the database of models, and a page is presented which allows the user to restrict the models to be displayed by organism type, model category and sub-category, and model author.
2) A request is sent to a Python script on the web server, which selects those models from the database, which satisfy the chosen criteria. These are then returned to the browser, which displays the list of possible models.
3) The user selects the model to run, and a request for the model details is sent to the server.
4) The webMathematica kernel manager allocates a Mathematica kernel.
5) The Mathematica kernel looks up the model details in a package file.
6) These are then passed to a Java applet, which is downloaded to the browser.

7) The applet sends a request to the webMathematica kernel manager to evaluate the model.
8) The evaluation request is sent to the Mathematica kernel, and on completion the results are returned to the kernel manager.
9) Finally, the results of the computation are sent to and displayed on the client machine.

JWS Online functionality

Interaction with JWS Online is done through a graphical user interface (GUI). A screen shot and two result windows are shown in Fig. 2. The interface consists of a number of panels where the user can make changes to the default parameter values of the model (Fig. 2, A), control the type of analysis that is required (Fig. 2, B), view a scheme of the model (Fig. 2, C) and its rate equations (Fig. 2, F) by moving the mouse over the red ovals in the scheme. Results are shown in separate windows (Fig. 2, D and E) depending on the type of simulation that is selected. In panel B the user can select for 1) a time simulation (Fig. 2, arrow 1), giving the options to plot either metabolite concentrations or flux values, 2) a steady-state analysis (Fig. 2, arrow 2), giving the options to do different types of structural analyses or to analyse for the steady-state solution, or 3) to do a Metabolic Control Analysis (Fig. 2, arrow 3), giving the options to either calculate the control coefficients or the elasticity coefficients. After selecting an analysis type the user can evaluate the model by clicking the Evaluate button (Fig. 2, arrow 4) and the results will be shown in a separate window. Examples of results windows are shown for a time simulation (Fig. 2, D) and a MCA analysis for control coefficients (Fig. 2, E).

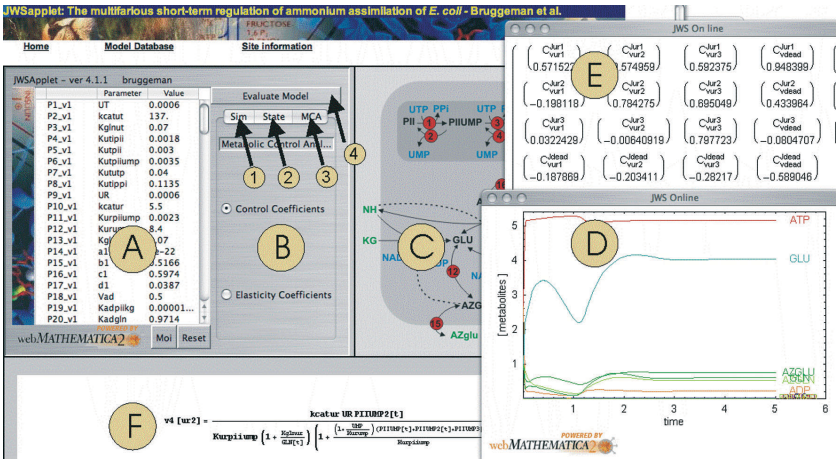

Figure 2. Screen shot of the JWS Online user interface and result windows. A screen shot is made of the JWS Online implementation of the model for the regulation of ammonia assimilation in *Escherichia coli* [23]. The Interface consists of different panels, A,B,C,F that allow control over the simulation and give information on the model (see text for details). In addition two result windows are shown (panel D and E) displaying a time simulation and an MCA result respectively.

The JWS Online team

Initially JWS Online was started in 2000 by Jacky Snoep and Brett Olivier as a challenge to see whether we could run Mathematica simulations over the internet. At that time web-Mathematica had not been developed and we used JLink to connect Java and Mathematica. JWS Online was launched in 2000 and has subsequently been significantly improved in a number of steps, the last one being the conversion to webMathematica on which we report here and for which Cor Stoof did the necessary Java programming.

At present Jacky Snoep is the PI of the JWS Online project with Carel van Gend as full time programmer. On a part-time basis Brett Olivier maintains the web site and Riaan Conradie, Franco Du Preez and Du Toit Schabort assist in coding models for the repository, Gerald Penkler and Kora Holm draw the metabolic schemes and make literature searches for manuscripts containing models.

Other Initiatives

Here we give a brief description of some other initiatives that have overlapping functionality with JWS Online. We have only listed initiatives that provide a repository of kinetic models for biological systems that are accessible via the internet (Table 1).

Table 1. A comparison between several web-based initiatives that store kinetic models and/or make models available for simulation. The initiatives are compared on their functionality with respect to whether they allow simulations to be run on the site (simulation), whether they store a collection of models (repository), whether the stored models are curated (i.e. do the models show the same behaviour as the published model, curation), whether the models are annotated (annotation) and whether the initiative is actively busy to add more models (here copying from other initiatives is not considered active, addition).

Initiative	URL	Simulator	Curation	Annotation	Addition
JWS Online	http://jjj.biochem.sun.ac.za	Yes	Yes	No	Yes
Virtual Cell	http://www.nrcam.uchc.edu	Yes	No	No	No
Biomodels	http://www.ebi.ac.uk/biomodels	No	Yes	Yes	Yes
WebCell	http://webcell.kaist.ac.kr	Yes	No	No	No
CellML	http://www.cellml.org	No	No	No	Yes
SBML	http://sbml.org	No	Yes	No	Yes
DOQCS	http://doqcs.ncbs.res.in/	No	Yes	Yes	Yes
ModelDB	http://senselab.med.yale.edu/senselab/ModelDB/	No	Yes	No	Yes
JSim	http://nsr.bioeng.washington.edu/	Yes	No	No	Yes
SigPath	http://www.sigpath.org/	No	No	Yes	No

The Virtual Cell [11,12] is hosted at the National Resource for Cell Analysis and Modeling (NRCAM) at the University of Connecticut Health Center and is a computational environment that helps in the construction and simulation of models that are cast in terms of ODEs or PDEs. The Virtual Cell follows a client–server set-up running Java applets;

clients can store models in a repository and import/export facilities for SBML, CellML and Matlab exist. The models are not curated or annotated (the client is responsible) and the Virtual Cell team does not actively add models to the repository.

The **Biomodels** [13] database is hosted at EMBL-EBI (UK) and is a collaborative effort between this institute, the SBML team (U.S.A.), the Systems Biology Group of the Keck Graduate Institute (U.S.A.), the Systems Biology Institute (Japan) and JWS Online. Biomodels focuses on model curation, annotation and import/export formats of published models. Models are curated to ensure that the published results can be reproduced. In the annotation process model components are linked to controlled vocabularies and other data resources. Models to be included in the database must be compliant with MIRIAM standards [14]. Both the Biomodels and JWS Online project are actively involved in adding models to their databases and these models are exchanged in SBML format between the two initiatives.

DOQCS [15] is hosted at The National Centre for Biological Sciences (NCBS) and is part of the Tata Institure of Fundamental Research in Bangalore. The Database of Quantitative Cellular Signaling is a repository of models of signalling pathways. It includes reaction schemes, concentrations, rate constants, as well as annotations on the models. The database provides a range of search, navigation and comparison functions. Export of models is available in GENESIS [16] and MATLAB (http://www.mathworks.com) format.

The **CellML** [17] and former **SBML** repositories hosted at the University of Auckland and CalTech respectively are repositories of kinetic models in XML format. The two modelling languages have significant overlap, CellML is aiming more at describing systems at the cellular level while SBML is better geared for reaction pathway models. CellML appears to have more freedom to define entities as components but is not as widely accepted as a format in simulation software. The models of the SBML repository have been improved and incorporated into BioModels Database.

SigPath [18] is hosted at the Weill Medical College of Cornell University, and at the Mount Sinai School of Medicine, it is an information management system designed to support quantitative studies on the signalling pathways and networks of the cell. SigPath focuses on storing, curating and annotating of quantitative information concerning signalling pathways. This information can be manipulated and reactions can be linked to form kinetic models. Some of these models (which are not curated as such) are available as a repository and can be exported in a number of formats amongst which SBML.

ModelDB [19] is a repository for published models from the neurosciences, it is part of the SenseLab project and hosted at Yale University. The models are available in the format in which it was submitted to the database (e. g. Fortran, NEURON).

JSim [20] is a simulation environment that can be used for the construction of models, a selected number of models is also available as Java applets and can be run over the web. JSim is closely linked to the NSR Physiome project, which provides comprehensive and downloadable physiological models [21].

WebCell [22] is hosted at the Korea Advanced Institute of Science and Technology (KAIST) and uses a client–server set-up with Java Servlet Pages and applets. New models can be added by the clients and stored in the database. The current models in the database are taken from the JWS Online, Biomodels and SBML repositories. The simulation functionality is similar to JWS Online.

Acknowledgements

The National Bioinformatics Network (NBN; South Africa) has provided funding for the infrastructure of the Stellenbosch node, including the JWS Server. In addition, C.v.G., B.G.O., R.C., and F.D.P. receive generous support from the NBN.

References

[1] Wolfram Research, Inc., Mathematica, Version 5.2, Champaign, IL (2005).

[2] http://www.postgresql.org/, PostgreSQL, Version 8.1.

[3] Chance, B., Garfinkel, D., Higgins, J., Hess, B. (1960) Metabolic control mechanisms V. A solution for the equations representing interaction between glycolysis and respiration in ascites tumor cells. *J. Biol. Chem.* **235:**2426–2439.

[4] Snoep, J.L., Westerhoff, H.V. (2004) The silicon cell initiative. *Current Genomics* **5:**687–697.

[5] Snoep, J.L. (2005) The SiliconCell initiative: working towards a detailed kinetic description at the cellular level. *Current Opin. Biotechnol.* **16:**336–343.

[6] Hucka, M., Finney, A., Sauro, H.M., Bolouri, H, Doyle, J.C., Kitano, H. *et al*. (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* **19:**524–531.

[7] Olivier, B.G., Snoep, J.L. (2004) Web-based kinetic modelling using JWS Online. *Bioinformatics* **20:**2143–2144.

[8] Snoep, J.L., Olivier, B.G., Westerhoff, H.V. (2004) JWS Online cellular systems modeling and the silicon cell. In: *Experimental Standard Conditions of Enzyme Characterizations.* (Hicks, M.G., Kettner, C. Eds), pp. 129–143. Logos Verlag, Berlin, Germany.

[9] http://tomcat.apache.org, Apache Tomcat, Version 5.5

[10] http://www.python.org, Python, Version 2.4

[11] Loew, L.M., Schaff, J.C. (2001) The Virtual Cell: a software environment for computational cell biology. *Trends Biotechnol.* **19:**401–406.

[12] Slepchenko, B.M., Schaff, J.C., Macara, I., Loew, L.M. (2003) Quantitative cell biology with the Virtual Cell. *Trends Cell Biol.* **13:**570–576.

[13] Le Novère, N., Bornstein, B., Broicher, A., Courtot, M., Donizelli, M., Dharuri, H., Li, L., Sauro, H., Schilstra, M., Shapiro, B., Snoep, J.L., Hucka, M. (2006) BioModels Database: A free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Res.* **34:**D689–D691.

[14] Le Novère, N., Finney, A., Hucka, M., Bhalla, U., Campagne, F., Collado-Vides, J., Crampin, E., Halstead, M., Klipp, E., Mendes, P., Nielsen, P., Sauro, H., Shapiro, B., Snoep, J.L., Spence, H.D., Wanner, B.L. (2005) Minimal information requested in the annotation of biochemical models (MIRIAM). *Nature Biotechnol.* **23:**1509–1515.

[15] Sivakumaran, S., Hariharaputran, S., Mishra, J., Bhalla, U.S. (2003) The Database of Quantitative Cellular Signaling: management and analysis of chemical kinetic models of signaling networks. *Bioinformatics* **19:**408–415.

[16] Bhalla, U.S. (2002) Use of Kinetikit and GENESIS for modeling signaling pathways. *Method. Enzymol.* **345:**3**–**23.

[17] Lloyd, C.M., Halstead, M.D.B., Nielsen, P.F. (2004) CellML: its future, present and past. *Progr. Biophys. Molec. Biol.* **85:**433–450.

[18] Campagne, F., Neves, S., Chang, C., Skrabanek, L., Ram, P.T., Iyengar, R., Weinstein, H. (2004) Quantitative Information Management for the Biochemical Computation of Cellular Networks, Sci. STKE 248, PL11.

[19] Migliore, M., Morse, T.M., Davison, A.P., Marenco, L., Shepherd, G.M., Hines, M.L. (2003) ModelDB Making models publicly accessible to support computational neuroscience. *Neuroinformatics* **1:**135–140.

[20] http://nsr.bioeng.washington.edu, Jsim

[21] http://www.physiome.org/Models/

[22] Lee, D.-Y., Yun, C., Cho, A., Hou, B.K., Park, S., Lee, S.U. (2006) WebCell: a web-based environment for kinetic modeling and dynamic simulation of cellular networks. *Bioinformatics* **22:**1150–1151.

[23] Bruggeman, F.J., Boogerd, F.C., Westerhoff, H.V. (2005) The multifarious short-term regulation of ammonium assimilation of Escherichia coli: dissection using an in silico replica. *FEBS J.* **272:**1965–1985.