

Modelica language – a promising tool for publishing and sharing biomedical models

Jiří Kofránek¹, Filip Ježek¹, Marek Mateják¹,

¹Department of Pathophysiology, 1st Faculty of Medicine, Charles University, Czechia
{kofranek, matejak.marek, jezekf}@gmail.com

Abstract

Current biomedical models are so extensive that their description (and reproducibility) requires more than a set of equations. Journal papers are thus frequently accompanied by electronic enclosures with detailed model descriptions, or even better, with a complete model source code. Specific electronic archives associated with specific languages and publicly accessible simulation platforms for the creation and archiving of biomedical models have been set up, however each of them has some disadvantage and an agreement on a common language for model sharing is missing. This paper reviews the usage of the languages for physiological modeling and discusses the advantages of the Modelica language in the area of physiological simulations.

Keywords: Physiology, Integrative models, Physiome project, Biomedical models archiving, Biomedical models publishing

1 The origin - a web of physiological regulations

In 1972 the medical journal Annual Review of Physiology published a paper (A. C. Guyton, Coleman, & Granger, 1972) which, at first glance, was absolutely different from typical physiological papers published at those times. A substantial part of the paper consisted of an extensive layout in a pasted-in enclosure. The layout, full of lines and interconnected elements, reminded slightly of the layouts of electric systems (Fig. 1). However, instead of electrical components, the layout displayed interconnected computer blocks (multipliers, dividers, adders, integrators, functional blocks) representing mathematical operations on physiological quantities (Fig. 2).

Bundles of connecting conductors between the blocks expressed, at first glance, complex feedback connection of physiological quantities. The blocks were grouped into 18 units representing interconnected physiological subsystems (Fig. 3). This was an entirely novel approach to the description of physiological regulations of the circulatory system and its broader physiological context and links to the other body subsystems – kidneys, regulation of volume and electrolyte equilibrium, etc., by means of graphical mathematical symbols. The complex systems of mathematical equations were replaced by a graphical representation of the mathematical relations. This

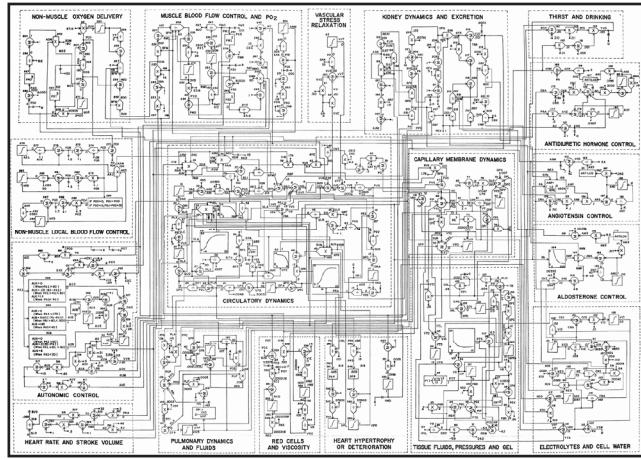


Figure 1. Diagram of Guyton's model (Guyton et al., 1972).

syntax allowed connections between the various physiological quantities to be represented by interconnected blocks standing for mathematical operations. Thus the entire layout was a formalized description of physiological interrelations in the circulatory system using a graphically represented mathematical model. The model description itself consisted mainly of a basic (still fully illustrative) picture. Any comments and justifications of the mathematical relation formulations were brief only. This required the reader (having a solid physiological and mathematical background) to fully concentrate in order to gain an understanding of the meaning of the formalized relations between/among the physiological quantities.

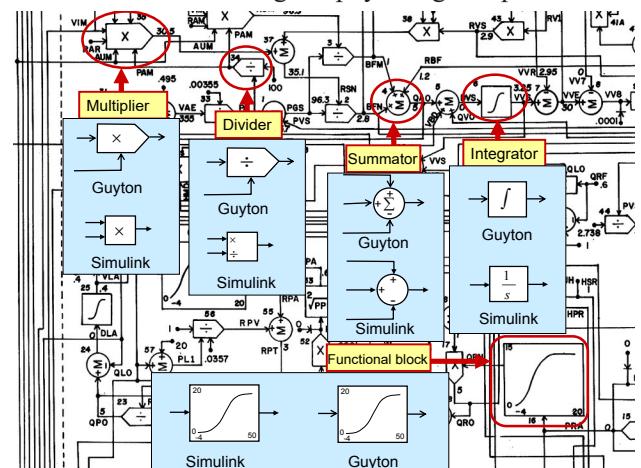


Figure 2. Individual elements in the scheme of the Guyton's model represent mathematical operations whose connections represent graphically expressed mathematical equations. Blocks in the original Guyton notation (1972), and the same blocks in Simulink (1990).

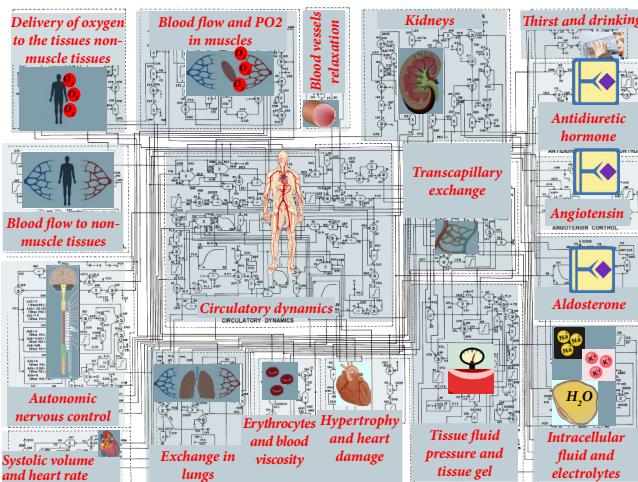


Figure 3. Interconnected physiological subsystems in the Guyton's model (Guyton et al., 1972).

A monography (Arthur C. Guyton, Jones, & Coleman, 1973) in which many of the approaches were explained in more detail was published a year later and a further monograph, by Guyton et al. (Arthur C. Guyton, Taylor, & Granger, 1975), presenting a reasonably detailed explanation of the mathematical formalization of the body fluid dynamics description, appeared in 1975.

2. Formalization of physiological relations – the PHYSIOME Project

Guyton's model was a milestone of sorts, applying a system approach to physiological regulations and describing the dynamics of interrelations between/among physiological subsystems by means of a system of graphically represented mathematical equations. Guyton's graphical diagram marked the emergence of an area of physiological research into the interconnected physiological systems in the living body, now referred to as “integrative physiology” (Coleman & Summers, 1997; Mangourova, Ringwood, & Van Vliet, 2011; Reinhardt & Seeliger, 2000).

Much as how theoretical physics strives to describe physical reality and explain the results of experimental research, “integrative physiology” tries, based on experimental results, to set up a formalized description of the interconnections of physiological regulations and explain their function both in the healthy body and during the development of various diseases.

Formalization, i.e. replacement of a verbal description of physiological systems with the precise language of mathematics, is closely linked to the issue of computer modeling. It is an asset of the formal description that deductions regarding the behavior of a system described by formalized tools are made based on the rules of a formalized language, i.e. by solving the equations of a mathematical model. This is a task that can be left to a computer – the computer solves equations describing the biological reality – and so it is computer simulation that is involved.

The concept of formalization started later and progress is somewhat slower in biological and medical sciences than in physics, chemistry and technology, because biological systems are much more complex. While the formalization process in physics started as early as the 17th century, formalization in medical and biological sciences came only together with cybernetics and computer science. This field of science uses computer models set up based on a mathematical description of the biological reality.

Formalized description of physiological systems is currently the subject of the international **PHysiome** Project (<http://www.physiome.org>), successor to the **GENOME** Project whose outcome consisted in a detailed description of the human genome. The aim of the **PHysiome** Project is to provide a formalized description of physiological functions (Bassingthwaite, 2000; P. Hunter, 2016; Peter J. Hunter, Crampin, & Nielsen, 2008; P. J. Hunter, Li, McCulloch, & Noble, 2006; P. Hunter, Robbins, & Noble, 2002; Omholt & Hunter, 2016). Physiome makes efforts to apply the formalized approach in order to integrate our knowledge, from the cell level to the organ level to the whole-body level, with a view to gaining insight into how all that works as a whole. The European initiative in this area is represented by the The International Union of Physiological Sciences (IUPS) <http://www.iups.org/physiome-project/>. The work of the IUPS Physiome Project has been boosted by the European Commission-funded **VIRTUAL PHYSIOLOGICAL HUMAN INITIATIVE** project (under the virtual physiological human institute <http://www.vph-institute.org/>), aiming, among other things, to apply the formalized approach to human physiology in clinical medicine and to use computer models in pre-clinical trials.

Integrative models of laboratory animals have been developing lately in addition to the integrative models of human physiology. For example, the aim of the **VIRTUAL RAT** project (<http://www.virtualrat.org/>) is to set up a complex model of the laboratory rat, which can readily be validated against experimental data on laboratory animals (Beard et al., 2012).

3. New modeling environments

In the meantime, general software simulation environments emerged, enabling models to be developed in a graphical format and allowing them to be debugged and ultimately verified. Among them is the widely used Matlab/Simulink tool from Mathworks, enabling a simulation model to be composed from various pre-defined components visually by drag-and-drop into the simulation networks. The Simulink blocks are very similar to the elements used by Guyton for a formalized representation of physiological relations. They actually differ in the graphical format only.

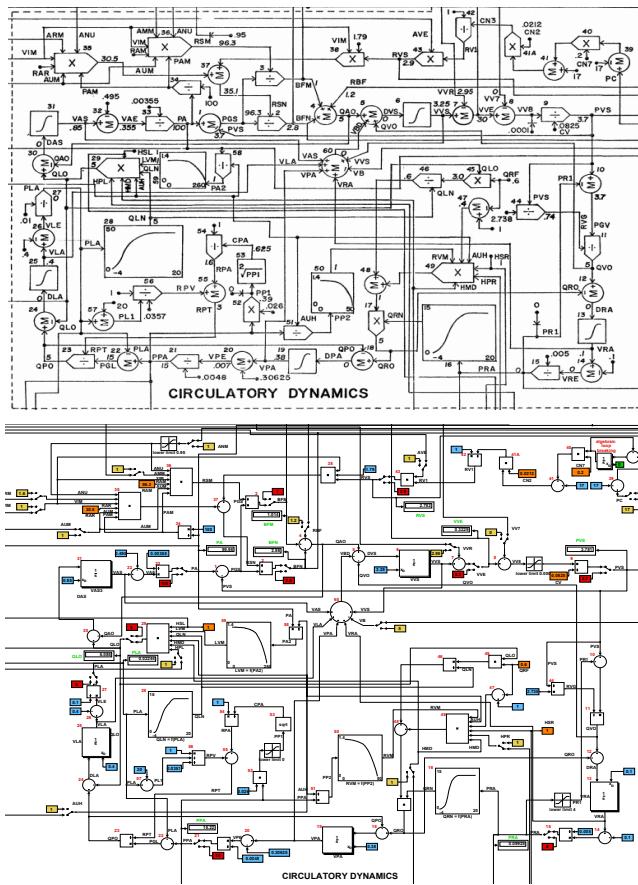


Figure 4. Circulatory dynamics - detailed representation of the central structure of the Guyton model in the original graphical notation (upper part of the figure) and in our Simulink implementation (bottom of the figure), which shows blood flows through aggregated parts of the circulatory system, and action of the heart as a pump (Jiří Kofranek & Rusz, 2010).

This similarity of Guyton's approach and philosophy of Simulink software inspired us to revive the old traditional Guyton's diagram by means of Simulink and transform it into a functional simulation model (Jiří Kofranek & Rusz, 2010). The appearance of the Simulink model could be nearly identical with that of the initial layout (Fig. 4).

Simulation visualization contained a number of errors (or rather “graphical typos”) in the initial layout. This poses no problem in a drawn picture, but the moment you try to revive it in Simulink, the model collapses as a whole immediately.

4. Model presentation in scientific publications

Guyton's diagram is just an illustrative picture condensing a system of equations describing a complex model into a graphic form. Since the description contained errors, it was difficult to reproduce the model based on the graphical diagram only. However, the authors made the model program in Fortran available on request, which en-

abled the behavior of the model to be tested.

This however introduces additional problem in versioning - the published version may produce different results than the later, usually further updated, provided on request.

It is a generally adopted principle that if a result is described in a scientific journal, then the experimental design must be reproducible at another workplace. The reproducibility principle plays a key role in scientists' efforts to disclose the secrets of Nature.

Actually, however, it is often violated in scientific publications dealing with biomedical models. This is not always a mistake of the authors – frequently just some letter or index is omitted, and it is then very difficult for the reader to understand the model or even to implement it.

The reviewers do not reimplement the models from the description (as they usually have the underlying code accessible on request) and thus the equations could easily contain a mistake.

Also, biomedical models are often so complex that the limited space allocated for the paper allows the authors just to present the basic model equations (and sometimes not all of them) and no space remains for additional information (starting values of variables of state, all parameter values, solver settings etc.) that is needed to set up the model at another workplace. Also, a number of articles do synthesize multiple models together, be it an extension of their previous research or adopted from literature. The details of combining the old (and referenced) with the newly presented do often raise a number of issues.

From our teaching experience, around 80 % of models implemented based solely on a description in a published article were incomplete or contained some error, which makes the model unusable. Nielsen et al. also support our observation of difficult reproducibility (Nielsen, Nilsson, & Matheson, 2012). A scientific paper describing a model should thus be accompanied by a digital enclosure (accessible on the Internet) containing a detailed description of the model structure, including the values of all parameters and most of all containing a complete source code in a common, formal programming language, adequate for the reader to be able to run the model, reproduce the model results and to potentially use the model as a basis for their own work where appropriate. The sharing of the complete source code is becoming common practice and even a requirement in a number of journals publishing scientific papers on computer models, especially the open-access ones.

5. Repositories of biomedical system models

A serious obstacle arises if a model is published in a

modeling language requiring a commercial license (such as Matlab/Simulink by MathWorks), because the reader must be a licensee of the particular system to be able to just reproduce the model results.

This is why considerable efforts have been made within the international **PHYSIOME** Project to create simulation languages appropriate for describing biomedical models and saving them in specific databases – model repositories. Publicly available tools for creating and launching models programmed in such languages were also created in this context.

So, for instance, the **Virtual Cell** project (<http://vcell.org>) has been set up for visualization and simulation of the cell metabolism and cell signal paths. That project was developed by the Center for Cell Analysis & Modeling, at UConn Health, University of Connecticut (USA). Quite a large group of users exists now around that project. The Virtual Cell developmental environment is interlinked with a number of databases and with the list of diverse models. This environment works on the client-server principle.

The “**Bio Tapestry**” project of Caltech (California Institute of Technology), Eric Davidson’s laboratory, is designed for modeling regulatory gene networks (where expression of the various genes is blocked/activated by transcription factors which, in turn, result from the expression of other genes) (<http://www.biotapecstry.org>). Regulatory gene networks look sort of like status automata (gene expression depending on the presence of the relevant transcription factors) – the gene expression may result in the formation of a protein, which can also be a transcription factor. A gene network editor and simulator can visualize the stepwise changes in the expression of the various genes and, based on a comparison with the experimental data, help explain the complex processes taking place particularly during embryonic development. Once again, this tool is interlinked with electronic model archives and has its own user community.

Two large global centers maintaining extensive physiological model databases are currently involved in the **PHYSIOME** Project.

The first center (founded by Jim Bassingthwaigte) is administered by Washington University in Seattle and uses the specifically created **JSim language** for the model database(Butterworth, Jardine, Raymond, Neal, & Bassingthwaigte, 2013). A description of the language, installation sources and tutorials are available at: <http://www.physiome.org/jsim>.

The environment for the creation and launching of models written in JSim is based on Java, owing to which it can be easily installed on different platforms. This environment can be used to modify and launch models from an extensive model database: <http://www.physiome.org/>

jsim/models.

The other large physiological system model database is maintained by the University of Auckland, New Zealand (<https://unidirectory.auckland.ac.nz/profile/phun025>). Petr Hunter, founder of the database, has built a top-ranking workplace in New Zealand – halfway between America and Europe (<http://www.abi.auckland.ac.nz/en.html>).

This institution uses the **CellML language** (Cooling & Hunter, 2015; Cuellar et al., 2003; Garny et al., 2008; Lloyd, Lawson, Hunter, & Nielsen, 2008) to describe the models: <http://www.cellml.org>. The tools for browsing, creating and launching models in CellML are available at <https://www.cellml.org/tools>. A tool for converting from CellML to JSim also exists. OpenCell is a tool for CellML simulation: <https://www.cellml.org/tools/opencell>.A large database of models has been created in CellML and is available at: <https://models.physiomeproject.org>. The models were taken from the literature and reprogrammed into CellML (or JSim). Each model is accompanied by reasonably detailed documentation. A model downloaded from the database in CellML can be simulated in the OpenCell environment.

However, the development of specialized simulation tools is limited by the funding allocated for the physiological research.

6. Equation-based languages

Both JSim and CellML are causal, block-oriented languages. The same characteristics also applies to Simulink (from Mathworks), frequently used to model biomedical systems.

The main problem with block-oriented languages lies in the fact that a simulation network consisting of hierarchically connected blocks is a graphical representation of a chain of input value transformations to output values. This means that an exact algorithm for the calculation chain from the input values to the output values must be defined when creating a model.

As a consequence of the requirement of a fixed connection direction from the inputs to the outputs, **the connection of the blocks reflects the calculation procedure rather than the structure of the modeled reality itself**.

Where complex models are involved, deriving the causality of the calculation (i.e. deriving the algorithm for calculation of the output variables from the input variables) is by no means a straightforward task.

This problem is addressed by modern equation-based, or acausal, modeling languages. Unlike block-oriented languages, where the structure of the hierachic block connections represents more the calculation method than the reality being modeled. the structure of the models in Modelica reflects the very structure of the reality mod-

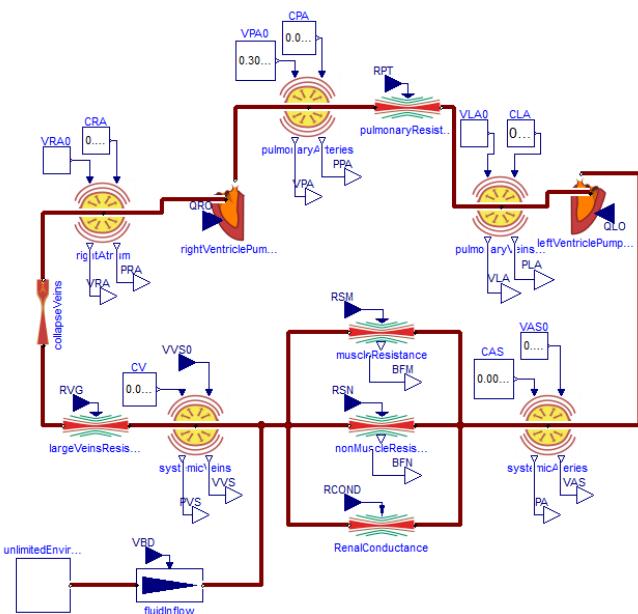


Figure 5. The same part of the model as in Figure 4, but implemented in Modelica. The model contains connected instances of two pumps (of the right and left heart ventricle), elastic vascular compartments, and resistances. Upon its comparison with Fig. 3, it can be seen that the model structure in Simulink corresponds rather to a computational algorithm, while the model structure in Modelica shows more of the structure itself of the modelled reality. Figure was adapted from (Kofránek, Mateják, & Privitzer, 2011).

eled (compare Fig. 4 in block-oriented language and Fig. 5 in Modelica). Owing to this, even complex models are adequately transparent and understandable in Modelica (Ježek, Kulhánek, Kalecký, & Kofránek, 2017).

A model should be **understandable** not only to the development team members but also to others. If only the authors understand their model, they will hardly obtain the necessary feedback or new impulses for their work from the scientific community.

This is of great importance with respect to the creation of complex integrative physiology models. When using block-oriented languages (be it Simulink or specifically created open-source languages for the documentation of biomedical models – JSim or CellML), the resulting complex program is not very comprehensible. It is largely only the authors who are able to understand their complex models. Modelica solves this problem efficiently, and integrative models of human physiology in Modelica have the potential for wider use within the scientific community.

Nowadays, the principles of equation-based approach are further implemented also in other products. E.g. SimScape (Mathworks, MA, USA) software package extends the commonly used Matlab/Simulink environment with the multidomain physical system modeling capabilities, useful also for biomedical engineering (de Canete, Saz-

7. Modelica – a language suitable for publishing and sharing biomedical models

Modelica, initially developed as an academic project in collaboration with small developmental companies at the universities in Lund and in Linköping, soon emerged as a highly effective and efficient tool for modeling complex models with potential application in mechanical engineering and in the automotive and aircraft industries.

Owing to this, the development of Modelica eventually gained support from the commercial sector, but the language itself is developed by an independent nonprofit association (see www.modelica.org). The Modelica Association gathers a number of key commercial as well as academic players, which ensures the stability of the platform and its relative independence on business decision of individual companies.

The speed at which this new simulation language spread to the various industries and was adopted by diverse commercial developmental environments is striking. Thanks to adoption by commercial sector, the language and both proprietary and open-source tools are already mature enough to guarantee reliable modeling platform. Several commercial as well as noncommercial developmental tools using this language currently exist (see www.modelica.org/tools).

Modelica users are therefore not confined to licensed commercial developmental tools: in fact, mature open-source developmental tools for this language exist now (e.g. OpenModelica, available at openmodelica.org, and JModelica, available at jmodelica.org/).

Thus, the effort spent on developing and maintaining own simulation platform is now unnecessary. Development driven by a number of high-tech industry (automotive, energy and aerospace) also guarantees small risk of stale development or platform discontinuation.

Modelica therefore appears to be a highly promising tool for publishing and sharing models. Some researchers have already adopted Modelica as their preferred modeling tool, e. g. (Heinke, Pereira, Leonhardt, & Walter, 2015; Maksuti, Bjällmark, & Broomé, 2015), even switched from the SimScape (de Canete, 2015), or use the general model exchange functional mockup interface, based on Modelica initiative (Gesenhus et al., 2017). The developers of the most complete physiological model HumMod are also considering using Modelica implementation to make the model easier to maintain (R. Hester, personal communication, August 2018). However, the penetration of Modelica in physiological research is still not massive.

| Connector: | | flow variable | nonflow variable |
|------------|------------|---|---|
| ■□ | Chemical | molar flow [mol.s ⁻¹] | concentration [mol.m ⁻³] |
| ◆◆ | Hydraulic | volumetric flow [m ³ .s ⁻¹] | pressure [Pa] |
| ■□ | Thermal | heat flow [W] | temperature [K] |
| ■□ | Osmotic | volumetric flow [m ³ .s ⁻¹] | osmolarity [mol.m ⁻³] |
| ■□ | Population | change [s ⁻¹] | size [I] |
| ■□ | Electrical | electric current [A] | electric potential [V] |

Table 1. Physical connectors in Physiolibrary compared with electrical connector in the Modelica Standard Library. Each connector in Physiolibrary defines one physical domain. As seen in Table 2, most of the components have analogies throughout the domains. For example, the resistor in electrical circuits has an analogy in the chemical domain as diffusion, because the molar flow of a substance is driven by the concentration gradient in the same way an electric current is driven by the voltage gradient.

8. Application libraries for biomedical simulations in Modelica

The proliferation of Modelica was facilitated by the existence of libraries for the most diverse areas, which appreciably simplify model formation for the given application domain. A model is set up by interconnecting instances of library components, like – figuratively speaking – buildings made of Lego bricks.

The majority of current libraries serve physical and technological applications. New libraries had to be created for models in the biomedical domain.

This is why we have created **Physiolibrary** (Marek Mateják et al., 2014), intended for model creation in physiology (see <http://physiolibrary.org>).

Physiology is a very progressive discipline, that examines how the living body works. And there is no surprise, that all processes in the human body are driven by physical laws of nature in several physical domain (see Tab 1 and Tab 2). And it is a great challenge to join many old empirical experiments with the ‘new’ physical principles. We hope, that this library helps the unflagging effort of Physiologists to exactly describe the processes and include their hypothesis.

The Physiolibrary contain basic physical laws in Human Physiology usable for cardiovascular circulation, metabolic processes, nutrients distributions, thermoregulation, gases transport, electrolytes and acid-base regulations, water distributions, hormonal or pharmacological regulations.

Chemical processes also have to be modeled in the bio-

| Resistance | Accumulation | Stream |
|---|----------------------------|--|
| $f_i = G^*(e_1 - e_2)$ $f_1 + f_2 = 0$ | $\int f = a$ $a = C^*e$ | $f_1 = \begin{cases} F e_1, & F \geq 0 \\ F e_2, & F < 0 \end{cases}$ $f_1 + f_2 = 0$ |
| G..conductance | C..capacitance | F..stream flow |
| | | |
| Chemical diffusion | Substance | Solution flow |
| | | not applicable |
| Hydraulic resistance | Elastic vessel | |
| | | |
| Heat convection | Heat | Heated mass flow |
| | | not applicable |
| Semipermeable membrane | Osmotic cell | |
| not applicable | | |
| | Population | Growth, Differentiation |
| | | not applicable |
| Electrical resistor | Electrical capacitor | |

Table 2. Analogies of selected Physiolibrary components based on connectors from Table 1 compared with electrical components in the Modelica Standard Library. To define the mathematical analogies in Table 2 we use the symbols e for effort (for connector nonflow variables) and f for flow (for connector flow variables). If there are more connectors in a component, they are differentiated by index. Unfortunately many elementary components in Physiolibrary do not have analogies through these domains. The special definitions in Physiolibrary include, for example, the components for chemical reaction, for hydrostatic pressure, for Henry’s solubility of gas in liquid, for Donnan’s equilibrium of electrolytes on membrane etc. Table 1 and 2 were presented in (M. Mateják & Kofránek, 2015).

medical area, and so we created the **Chemical** library as well (Matejak, Tribula, Ježek, & Kofranek, 2015).

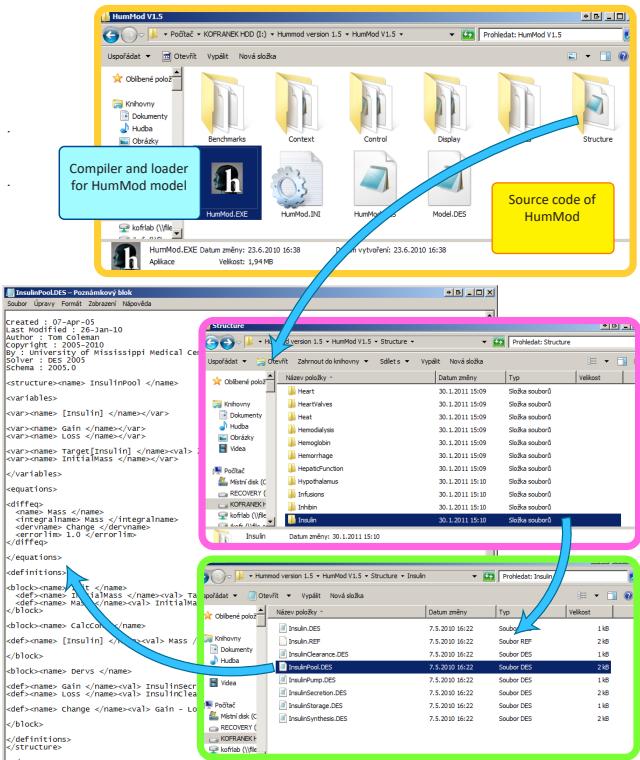


Figure 6. HumMod simulator has been distributed with a compiler, loader and the source code written in thousands of XML files.

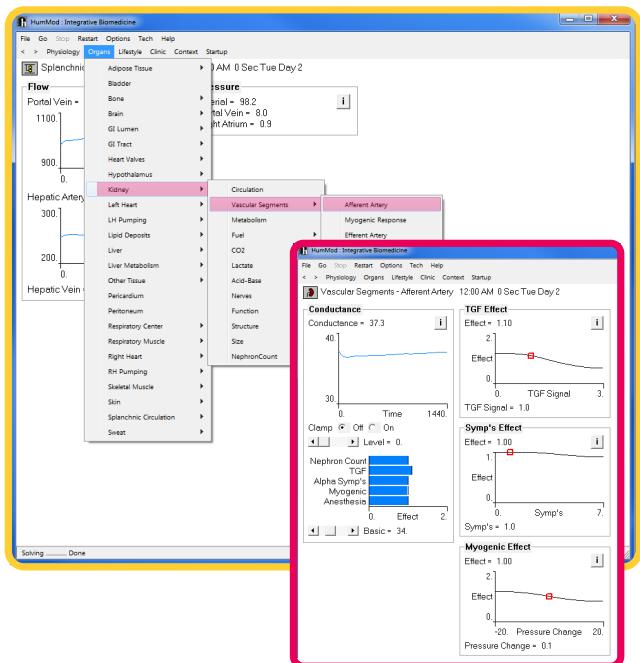


Figure 7. The user can compile and run the HumMod model. Using a widely branched menu, hundreds of variables can be monitored during simulation experiments.

Those libraries are a result of our many years' experience in the implementation of extensive hierarchical models of human physiology in Modelica (Ježek et al., 2017; Jiri Kofranek, Matejak, & Privitzer, 2011), the HumMod model in particular.

The **HumMod** model, set up in international collaboration by a group of collaborators and disciples of A. Guyton at the Mississippi University Medical Center, USA, (R. Hester, Brown, Husband, & Iliescu, 2011; R. L. Hester, Coleman, & Summers, 2008) is probably the most extensive existing model of integrated physiological systems of human physiology. The authors do not keep the structure secret: the model source text (containing over 5,000 variables) can be downloaded from the project web pages: <http://hummod.org>. The source text is written in a specific XML markup language. The whole mathematical model is offered as an open-source tool. The user is free to download both the source text and the translator into their computer from the web page and to launch the model on their own computer (Fig. 6 and 7). The user is in a position to modify the model to suit their purpose. A problem is in the fact that the XML source texts of the entire model are written in thousands of files located in hundreds of folders, and gaining insight into the mathematical relation by browsing through thousands of interlinked XML files is very difficult.

It appears that the **comprehensibility of the descriptions of complex integrative models** is one of the factors limiting their adoption by the scientific community. If the creators are the only ones to understand their model, any possibility of technical communication with other scientists is considerably limited. And so is the potential for a wider use within the broad scientific community. So, the development of methodologies that will make the description of the structure of complex hierarchical models so clear that a wide group of users can understand it is gaining in importance.

Specific browsers allowing the relations in the model to be browsed have been created in order to facilitate understanding of the HumMod model (Wu, Chen, Pruett, & Hester, 2013). Even so, the equations in the model and their interrelations are rather difficult for the user to understand. One of the ways to make understanding complex hierarchic models easier is to use the Modelica language. This is why we decided to re-implement the entire complex model of the US authors in Modelica.

Model re-implementation in Modelica makes the model structure much clearer (see Fig. 8), the source code resembling hierarchic physiological schemes. Making the model clearer also helped detect some errors in the initial US implementation of the HumMod model. We modified HumMod and extended it mainly in the area of modeling blood gas transfer and homeostasis of the inner environment, the acid-base equilibrium in particular (Jiri Kofranek et al., 2011).

Our version of the HumMod model, called **Physiomodel**, is being developed as an open-source tool. The **model source texts** (i.e. equations, values of all constants, etc.), which constitute a formalized representation of the

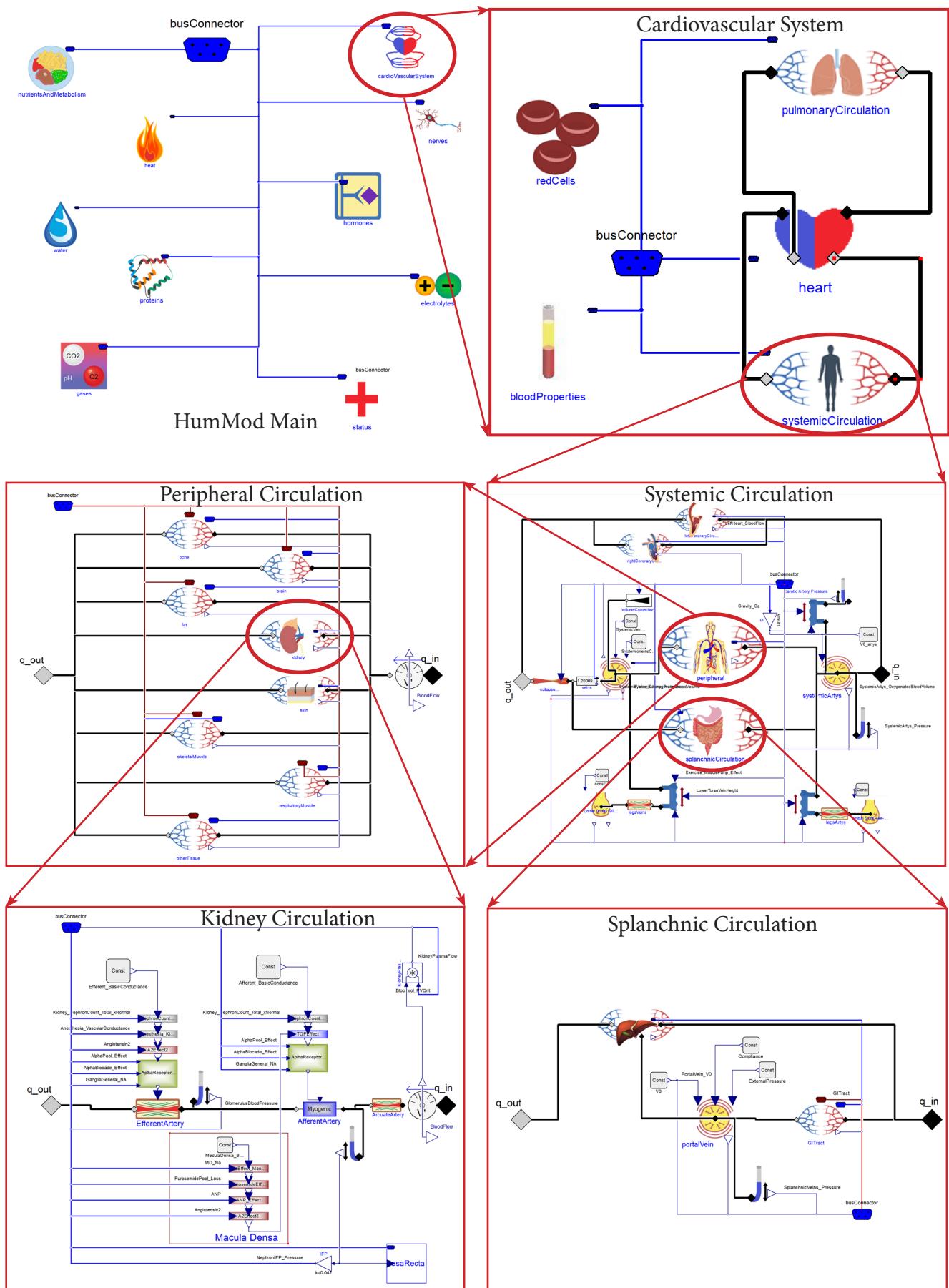


Figure 8. Illustration of a part of the source text of our HumMod implementation in Modelica. The source text resembles hierarchical physiological schemes. Image adapted from <http://www.physiomodel.org/>

physiological relations, are *publicly available* at <http://www.physiomodel.org>.

9. Community for biomedical model evolution in Modelica

International cooperation in combination with the openness of result sharing is a driving force of scientific development in today's globalized world. As experience shows, the development of complex software systems is conditional on the existence of the widest possible user/developer community to provide feedback and ensure further innovation of the complex products through cooperative efforts, with the ultimate use in commercial applications. This is why the development of projects with open source codes has become so popular lately.

The creation of extensive models of physiological systems is not just a purely scientific issue: they also potentially can be used in commercial applications. Medical simulators, which are based on a validated human physiology model (like aircraft simulators are based on an adequately faithful aircraft model), are a good example.

The development of complex integrated physiology models may be optimal if open scientific development is combined with the exploitation of business opportunities and financing by the commercial sector.

The creation of *OpenModelica* (<https://openmodelica.org>) within an open community may serve as a model in this area. The development of the products is managed by a consortium joining together universities, commercial companies and individual developers (*Open Source Modelica Consortium* - see <https://openmodelica.org/home/consortium>). The consortium members include both large companies and small developmental enterprises. Research is funded from member fees, the height of which depend on the company size as well as on the number of sold products developed by using the OpenModelica licenses. A reasonably large community combining users with many cooperating developers has concentrated around OpenModelica, resulting in a well-performing open-source product which is competitive in the existing environment of expensive commercial Modelica implementations. Commercial companies are free to use any part of the OpenModelica environment and expand it as appropriate, also during the development of competitive commercial implementations of the Modelica language (this is why companies such as Wolfram and MapleSoft have joined the consortium).

It is conceivable that an association of the academic community and commercial companies built on similar foundations and called, say, "*Physiomodelica Open Source Consortium*", may ensure further development of integrative physiology models in the future.

8 Conclusions

Let us sum up the factors owing to which Modelica is a language suitable for publishing and sharing biomedical models:

1. Modelica is a modeling language, not a proprietary product owned by a commercial company (such as, e.g. Mathworks' Matlab and Simulink).
2. Publicly accessible noncommercial developmental tools (such as OpenModelica and JModelica) exist for Modelica and are mature and reliable enough, the development is driven by well funded industries.
3. Modelica includes application libraries facilitating biomedical system modeling.
4. The model structure in the acausal Modelica language is clear, reflecting more the structure of the original modeled than that of the calculation and enabling extensive hierachic models to be set up.
5. Modelica may be broadly used in a number of application domains. Further Modelica developments are aimed at satisfying the requirements of the industries and are not dependent on grant funds from the PHYSIOME Project.

Acknowledgements

The authors appreciate the partial funding of this work by PRVOUK P/24/LF1 and MPO FV20628

References

- Bassingthwaite, J. B. (2000). Strategies for the physiome project. *Annals of Biomedical Engineering*, 28(8), 1043–1058.
- Beard, D. A., Neal, M. L., Tabesh-Saleki, N., Thompson, C. T., Bassingthwaite, J. B., Shimoyama, M., & Carlson, B. E. (2012). Multiscale modeling and data integration in the virtual physiological rat project. *Annals of Biomedical Engineering*, 40(11), 2365–2378.
- Butterworth, E., Jardine, B. E., Raymond, G. M., Neal, M. L., & Bassingthwaite, J. B. (2013). JSim, an open-source modeling system for data analysis. *F1000Research*, 2, 288.
- Coleman, T. G., & Summers, R. L. (1997). Using mathematical models to better understand integrative physiology. *Journal of Physiology and Biochemistry*, 53, 45–46.
- Cooling, M. T., & Hunter, P. (2015). The CellML Metadata Framework 2.0 Specification. *Journal of Integrative Bioinformatics*, 12(2), 260.
- Cuellar, A. A., Lloyd, C. M., Nielsen, P. F., Bullivant, D. P., Nickerson, D. P., & Hunter, P. J. (2003). An Overview of CellML 1.1, a Biological Model Description Language. *Simulation*, 79(12), 740–747.
- de Canete, J. F. (2015). Object-Oriented Programming for Modeling and Simulation of Systems in Physiology. *International Journal of Medical, Health, Biomedical, Bioengineering and Pharmaceutical Engineering*, 9(4),

- 343–346.
- de Canete, J. F., Saz-Orozco, P. del, Moreno-Boza, D., & Duran-Venegas, E. (2013). Object-oriented modeling and simulation of the closed loop cardiovascular system by using SIMSCAPE. *Computers in Biology and Medicine*, 43(4), 323–333.
- Garny, A., Nickerson, D. P., Cooper, J., Weber dos Santos, R., Miller, A. K., McKeever, S., ... Hunter, P. J. (2008). CellML and associated tools and techniques. *Philosophical Transactions. Series A, Mathematical, Physical, and Engineering Sciences*, 366(1878), 3017–3043.
- Gesenhues, J., Hein, M., Ketelhut, M., Habigt, M., Rüschen, D., Mechelinck, M., ... Abel, D. (2017). Benefits of object-oriented models and ModeliChart: modern tools and methods for the interdisciplinary research on smart biomedical technology. *Biomedizinische Technik. Biomedical Engineering*, 62(2), 111–121.
- Guyton, A. C., Coleman, T. G., & Granger, H. J. (1972). Circulation: overall regulation. *Annual Review of Physiology*, 34, 13–46.
- Guyton, A. C., Jones, C. E., & Coleman, T. G. (1973). *Circulatory physiology: cardiac output and its regulation*. Philadelphia: WB Saunders, 1973.
- Guyton, A. C., Taylor, A. E., & Granger, H. J. (1975). *Circulatory Physiology II. Dynamics and control of the body fluids* (Vol. 2). Saunders.
- Heinke, S., Pereira, C., Leonhardt, S., & Walter, M. (2015). Modeling a healthy and a person with heart failure conditions using the object-oriented modeling environment Dymola. *Medical & Biological Engineering & Computing*, 53(10), 1049–1068.
- Hester, R., Brown, A., Husband, L., & Iliescu, R. (2011). HumMod: a modeling environment for the simulation of integrative human physiology. *Frontiers in Physiology*. Retrieved from <http://journal.frontiersin.org/article/10.3389/fphys.2011.00012>
- Hester, R. L., Coleman, T., & Summers, R. (2008). A multilevel open source integrative model of human physiology. *The FASEB Journal*, 22(1 Supplement), 756.8–756.8.
- Hunter, P. (2016). The Virtual Physiological Human: The Physiome Project Aims to Develop Reproducible, Multiscale Models for Clinical Practice. *IEEE Pulse*, 7(4), 36–42.
- Hunter, P. J., Crampin, E. J., & Nielsen, P. M. F. (2008). Bioinformatics, multiscale modeling and the IUPS Physiome Project. *Briefings in Bioinformatics*, 9(4), 333–343.
- Hunter, P. J., Li, W. W., McCulloch, A. D., & Noble, D. (2006). Multiscale modeling: physiome project standards, tools, and databases. *Computer*, 39(11), 48–54.
- Hunter, P., Robbins, P., & Noble, D. (2002). The IUPS human Physiome Project. *Pflugers Archiv: European Journal of Physiology*, 445(1), 1–9.
- Ježek, F., Kulhánek, T., Kalecký, K., & Kofránek, J. (2017). Lumped models of the cardiovascular system of various complexity. *Biocybernetics and Biomedical Engineering*, 37(4), 666–678.
- Kofránek, J., Mateják, M., & Privitzer, P. (2011). Complex model of integrated physiological systems - a theoretical basis for medical training simulators. *Mefanet Report*, 4, 22–59.
- Kofránek, J., Mateják, M., & Privitzer, P. (2011). Hummod-large scale physiological models in modelica. In *Proceedings of the 8th International Modelica Conference*; March 20th–22nd; Technical University; Dresden; Germany (pp. 713–724). Linköping University Electronic Press.
- Kofránek, J., & Rusz, J. (2010). Restoration of Guyton's diagram for regulation of the circulation as a basis for quantitative physiological model development. *Physiological Research*, 59(6), 897.
- Lloyd, C. M., Lawson, J. R., Hunter, P. J., & Nielsen, P. F. (2008). The CellML Model Repository. *Bioinformatics*, 24(18), 2122–2123.
- Maksuti, E., Bjällmark, A., & Broomé, M. (2015). Modelling the heart with the atrioventricular plane as a piston unit. *Medical Engineering & Physics*, 37(1), 87–92.
- Mangourova, V., Ringwood, J., & Van Vliet, B. (2011). Graphical simulation environments for modelling and simulation of integrative physiology. *Computer Methods and Programs in Biomedicine*, 102(3), 295–304.
- Mateják, M., & Kofránek, J. (2015). Physiomodel - an integrative physiology in Modelica. In 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (pp. 1464–1467).
- Mateják, M., Kulhánek, T., Šilar, J., Privitzer, P., Ježek, F., & Kofránek, J. (2014). Physiolibrary-Modelica library for physiology. In *Proceedings of the 10 th International Modelica Conference*; March 10-12; 2014; Lund; Sweden (pp. 499–505). Linköping University Electronic Press.
- Mateják, M., Tribula, M., Ježek, F., & Kofánek, J. (2015). Free Modelica Library for Chemical and Electrochemical Processes. In *Proceedings of the 11th International Modelica Conference*, Versailles, France, September 21–23, 2015 (pp. 359–366). Linköping University Electronic Press.
- Ngo, C., Dahlmanns, S., Vollmer, T., Misgeld, B., & Leonhardt, S. (2018). An object-oriented computational model to study cardiopulmonary hemodynamic interactions in humans. *Computer Methods and Programs in Biomedicine*, 159, 167–183.
- Nielsen, T. A., Nilsson, H., & Matheson, T. (2012). A formal mathematical framework for physiological observations, experiments and analyses. *Journal of the Royal Society Interface*, 9(70), 1040–1050.
- Omholt, S. W., & Hunter, P. J. (2016). The Human Physiome: a necessary key for the creative destruction of medicine. *Interface Focus*, 6(2), 20160003.
- Reinhardt, H. W., & Seeliger, E. (2000). Toward an Integrative Concept of Control of Total Body Sodium. *News in Physiological Sciences: An International Journal of Physiology Produced Jointly by the International Union of Physiological Sciences and the American Physiological Society*, 15, 319–325.
- Wu, K., Chen, J., Pruett, W. A., & Hester, R. L. (2013). Hummod browser: An exploratory visualization tool for the analysis of whole-body physiology simulation data. In *2013 IEEE Symposium on Biological Data Visualization (BioVis)* (pp. 97–104). ieeexplore.ieee.org.