International Journal of Biological, Life and Agricultural Sciences
ISSN: 2415-6612
Vol:1, No:10, 2007

Analysis and Prototyping of Biological Systems: the Abstract Biological Process Model

Antonio Di Leva, Roberto Berchi, Gianpiero Pescarmona, and Michele Sonnessa

***Abstract*—**The aim of a biological model is to understand the integrated structure and behavior of complex biological systems as a function of the underlying molecular networks to achieve simulation and forecast of their operation. Although several approaches have been introduced to take into account structural and environment related features, relatively little attention has been given to represent the behavior of biological systems. The Abstract Biological Process (ABP) model illustrated in this paper is an object-oriented model based on UML (the standard object-oriented language). Its main objective is to bring into focus the functional aspects of the biological system under analysis.

***Keywords*—**Biological processes, system dynamics, system modeling, UML.

I. Introduction

THE aim of a biological model is to understand the integrated structure and behavior of complex biological systems as a function of the underlying molecular networks to achieve simulation and forecast of their functioning. Although several approaches (e.g., biological ontologies [1]) have been introduced to take into account **structural** and **environment** related features, relatively little attention has been given to represent behavior of biological systems [2].

The main objective of the Abstract Biological Process (ABP) model that will be briefly illustrated in the rest of this paper is to take into focus the functional aspects of a biological system. The ABP model is an object-oriented model that is based on UML (the standard object-oriented language [3]). These models are now well understood and began to be used in biological system specification [4].

A relevant problem that UML presents as a tool for the representation of complex systems is related to the fact that UML is a discrete modeling language that provides no direct means for modeling continuous system behavior. However, object-orientation is not limited to using only one tool and provides many mechanisms to incorporate different sub-models that might have been built using different paradigms (see, e.g., the framework based on the methodology called *multifaceted modelling* introduced in [5]).

Specifically, new approaches have been discussed to integrate different formalisms in order to face the complexity of real life systems. For instance, in [6] an integration of UML and System Dynamics has been proposed to model business processes. System Dynamics is a powerful modelling paradigm that is able to specify both static and dynamic aspects of complex systems. Moreover, it provides a graphical representation layer (with *dependence diagrams* and *stock-flow diagrams*) related to a well-defined mathematical differential equations layer [7]. In the ABP approach, the System Dynamics is used in the first phase of the project, the knowledge elicitation process by which the behavior of the system is defined.

This paper is organised as follows. Section II introduces the basis of the Abstract Biological Process (ABP) model as a systemic modelling language for a semi-formal description of biological systems and processes. Section III illustrates the process of knowledge acquisition discussing an example related to the production of specific peptides. Section IV states some preliminary conclusions.

II. The Abstract Biological Process (ABP) Model

The ABP model is a model which is based on a functional view of the biological system under analysis. In this section we will give a very preliminary introduction to the model.

The ABP model is based on some starting assumptions:

1. the aim of a **biological system** is to reproduce itself and/or survive, according to the local conditions (molecules and energy supply);
2. a generic **biological system** is a set of generic functional **sub-systems** (or **apparatuses**) which have well defined objectives;
3. from a functional point of view, a biological system can be considered as a usually very complex set of concurrent **biological functions**;
4. biological functions are decomposed into **biological processes** that accomplish the transformations of the biological system to realize its objectives;
5. usually each biological process can be decomposed into lower-level biological processes. Thus, a biological system can be recursively defined by the single unifying biological process concept;

Manuscript received November 15, 2005.

A. Di Leva is with the Dipartimento di Informatica, Università di Torino, Torino, Italy (corresponding author - phone: +39-011-6706759; fax: +39-011-751603; e-mail: dileva@di.unito.it).

R. Berchi is with the Dipartimento di Statistica, Università "La Sapienza" Roma, Italy (e-mail: r.berchi@tiscali.it).

G. Pescarmona is with the Dipartimento di Genetica, Università di Torino, Torino, Italy (e-mail: gianpiero.pescarmona@unito.it).

M. Sonnessa is with the Dipartimento di Informatica, Università di Torino, Torino, Italy (e-mail: sonnessa@di.unito.it).

6. in the object-oriented system model proposed in this paper, the collection of biological process is generalized into what we call an **Abstract Biological Process (ABP) class**, and each **abstract biological process** is an instance of this class. This means that each internal susb-system represents an abstract function that an abstract biological process would have to perform.

As shown in Fig. 1, an ABP has several **ports**, which allow exchanges with different components of its environment.

Fig. 1 can be interpreted in the following way:

- a generic ABP under analysis (denoted as *ABP*) can be a sub-process of a Super-ABP and a super-process of a Sub-ABP; commands and feedbacks are exchanged between the *ABP* and its super- and sub-processes;
- the *ABP* can be a part of a complex process Comp-ABP and can be composed of several component processes Part_of-ABPs; commands and feedbacks are exchanged between the *ABP* and its composed and components processes;
- an *ABP* can send/receive signals to/from the Environment and Other-ABPs;
- an *ABP* exchanges products with Supplier- and Client-ABPs; Input management consists of internal functions which control input products, store them and distribute products to the various recipients within the *ABP*; Output management consists of internal functions which receive output products from internal services, control them and deliver products to the Client-ABPs;
- the Managent and Control function is the internal supervisor function of the *ABP*; it is responsible for servicing the ports that connect the *ABP* to the environment and its related ABPs; the principal activity of the Management and Control function is to collaborate in order to satisfy the goals of the complex ABP of which it is part of, as well as plans for their implementation using local services or delegating activities to its component ABPs;
- Services represent a set of functions internal to the *ABP*; a service receives products from upstream ABPs (via Input management functions) and signals from the Management and Control function, and transforms them into other signals and products to be delivered to client ABPs.

Obviously, the ABP model is very general and real processes may have not all the features which have been described. For instance, the ports in Fig. 1 are themselves abstractions that are use to support the encapsulation of the internal behaviour. Since ports describe exchanges, they can be modeled using well-known object-oriented system analysis techniques as *use cases* diagram generation.

In this paper we will not focus on the general model but we will illustrate a knowledge elicitation process by which the complexity of real life systems can be faced. This process is based on a form of prototyping which uses the System Dynamic formalism to indagate the behaviour of such systems.

III. Analysis of Biological Systems

The analysis of a complex system is, first of all, a knowledge elicitation process by which the behavior of the system is determined and then specified using a particular method (such as UML).

The two most important specification techniques in object-oriented analysis are class diagrams and use case diagrams. These are techniques for data and function specifications that have been applied, for instance, to specify static and dynamic relationships which occur between SARS-CoV and a cell [4].

In this paper we will focus on the system behavior determination phase because this is the basic activity to model in a consistent way the system under analysis. This activity can be a very difficult one when it is applied to biological systems, whose basic mechanisms are not fully understood. In our opinion, this uncertainty suggests the development of a prototype of the system.

Prototyping is a powerful method for requirements and knowledge elicitation. A prototype is a "demonstration" system that is constructed to "visualize" the system behavior to the experts in order to obtain their feedback.

Let us illustrate this approach for a very simple case. After a bone marrow transplantation (performed in order to cure a leukemia) the availability of a specific protein, Fibrinogen (FB), is the key factor for the replication of the cells that produce leucocytes (neutrophils).

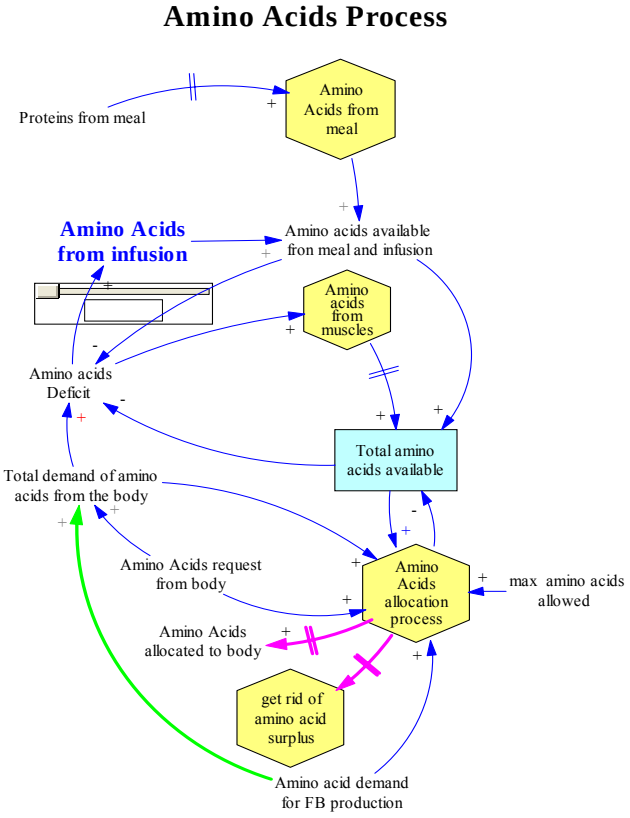

Fig. 2 The dependence diagram of the Amino Acid Process

Aminoacids have to be produced from meal (in this case, stomach, pancreas, intestine and liver perform the process), can be obtained from muscles, and/or can be provided by direct infusion. In Fig. 2 the process is represented as a "macro" influence diagram, the *dependence diagram*.

The hexagons represent elementary sub processes. The rectangle represents the amino acid *stock* available. The slider symbol reminds that the surgeon can decide to provide directly amino acid by direct infusion. The symbol *double bars* on the arcs tells us that there is a delay present (it take some time to get amino acid from a steak, it is the digestion time).

In the dependence diagram of Fig. 2 some feedback loops appear. The most important is the balancing loop that control the level of amino acids available vs the needed. As normally happens with this kind of models it is difficult to define the boundaries: amino acid are requested for the body equilibrium but they are also important for the production of the fibrinogen for the developing cells. An important point is the representation of the exogenous decision regarding the Amino Acid direct infusion.

In the System Dynamics approach, the *dependence diagram* has to be converted into a *stock-flow diagram* in which stock levels are governed by flows in and flows out. This type of diagram can be considered as a visual language for representing differential equation model, with a stock representing a state variable and the rate of change being the net of inflows minus outflows. The stock-flows diagram of the aminoacid production process is show in Fig. 3. We decide not to represent that part of model related to the acquisition of amino acids from muscles. This decision has been made because the time span of the model (30 days) is short and because direct infusion supersedes the problem.

In Fig. 4, a prototype of one of the possible control panels useful in determining the patient status and in forecasting his evolution is illustrated. In the pane blue section (lower right) are reported the measures related to the status of the patient and some physiological constants. In the bottom left the action that the doctor can perform (infusions of AA or Platelets). The graphs on the top show both the actual data and the forecasts coming from the model.

The fitting has been considered from the experts quite interesting, the model is able to reproduce the actual behavior and this is true for an optimal evolution of the post operation phase.

A counter example of a possibly bad evolution due to a malpractice could be simulated. Let us suppose that the platelets infusion is too high: the dangerous situation in which the patient could be brought is shown by means of the graphs in Fig. 5 which have been obtained simulating the model under this assumption. This situation is a dangerous one: there is a risk of thrombosis due to the too high level of Fibrinogen (and D-dimer) in the blood.

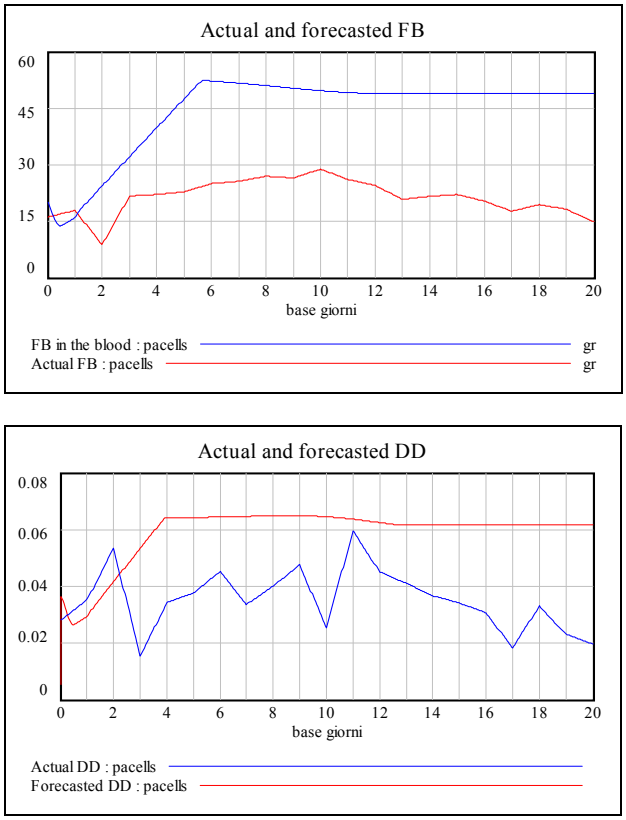

Fig. 5 "What-if" simulation

In practice the model could be used as a monitoring tool that is able to anticipate critical situation ("what-if" analysis). Naturally, the actual situation for each patient should be taken automatically from the patient data base and the model interface should be personalized to doctor's needs.

IV. Conclusion

The model has been built in order to explain the evolution of the situation after a bone marrow transplantation. Its top-down approach, starting from the macroscopic level, allow domain experts to easily understand what is included into the model (and what is excluded). This has an important influence on many aspects of model building , improving the development time of the model, the utilization and retrieval of information, the motivation of retrieving and storing actual information, and so on.

This preliminary model has demonstrated the utility of the System Dynamics approach to increase understanding, to provide a quick working model of the biological system, and to support the operators with a tool that can be used to monitor the patient's status and test the consequences of different actions in a simulation environment.

REFERENCES

[1] Stein, L. "Integrating biological databases" Nat. Rev. Genet., 4, 337-345, 2003.
[2] P. E. Midford "Ontologies for Behavior" BIOINFORMATICS, Vol. 20 no. 18, 2004, pp. 3700–3701.
[3] Rumbaugh, J., Jacobson, I. and Booch, G.: "The Unified Modeling Language Reference Manual"; Addison Wesley, 1999.
[4] D.Shegogue and W.J. Zheng "Object-oriented modeling of the SARS coronavirus" BIOINFORMATICS, Vol. 21 no. 10, 2005, pages 2502–2509.
[5] B.P. Zeigler, H. Praehofer, T.G. Kim Theory of Modeling and Simulation, 2nd Ed., Academic press, 2000.
[6] Chang, Liang-Cheng and Yi-Ming Tu "Attempt to Integrate System Dynamics and UML in Business Process Modeling" Proceedings The 23rd International Conference of the System Dynamics Society July 17-21, 2005 Boston.
[7] Sterman, J. Business Dynamics: Systems Thinking and Modeling for a Complex World, McGraw Hill, 2000.

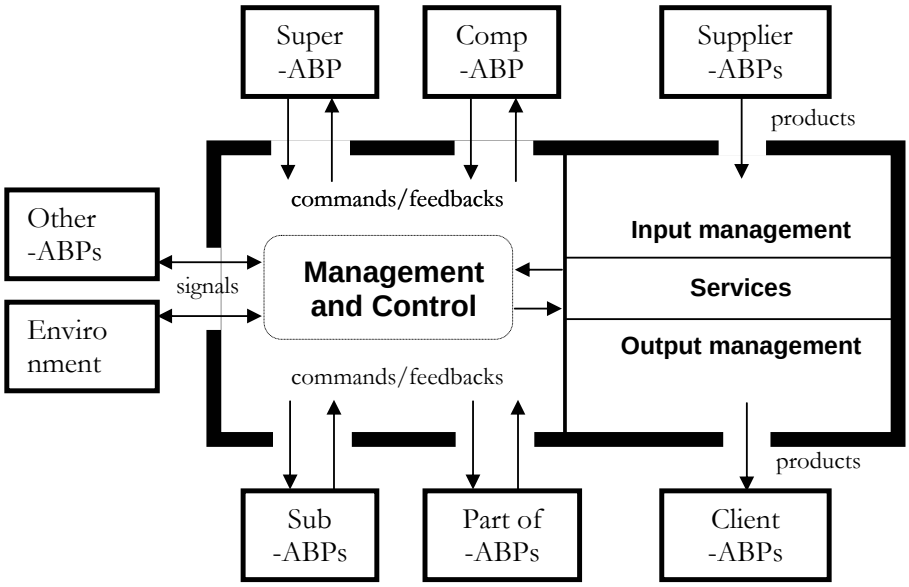

Fig. 1 The ABP model

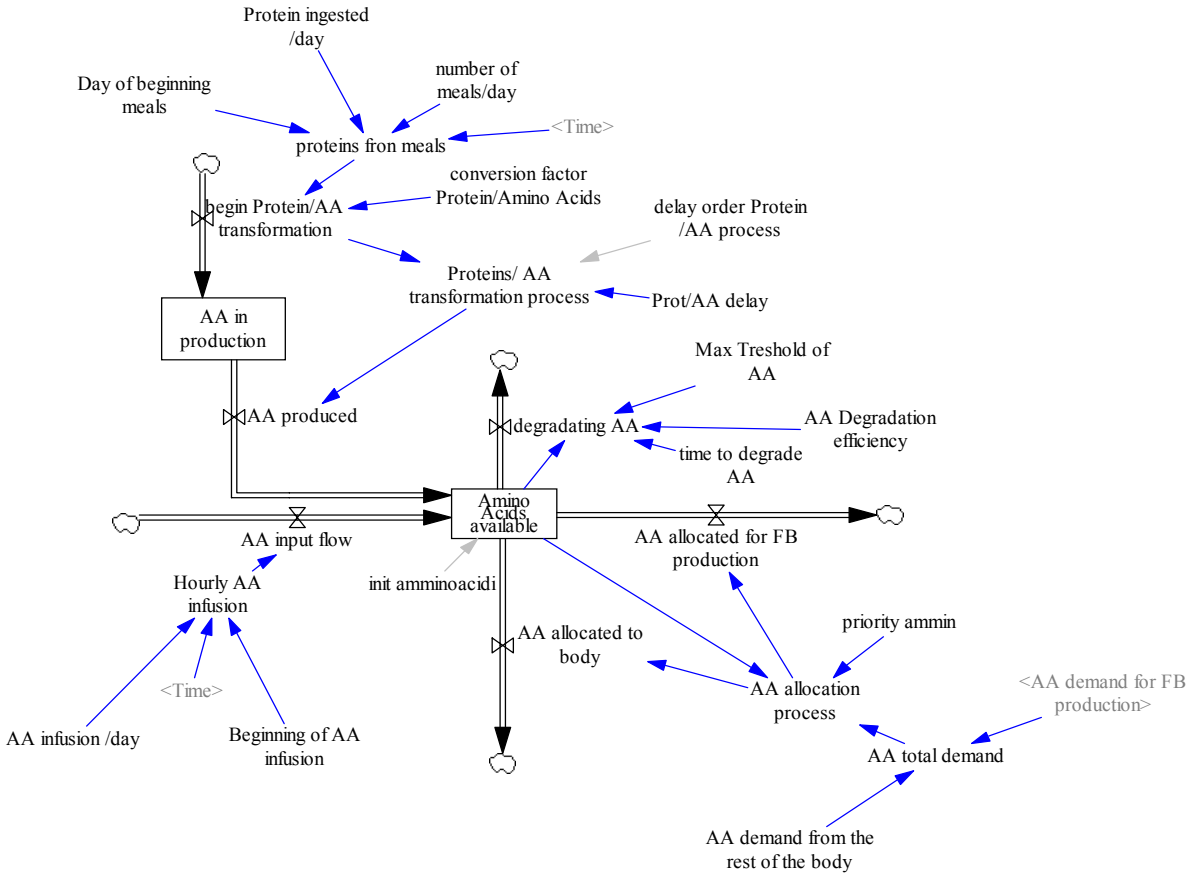

Fig. 3 The stock flow diagram of the Amino Acids Production process

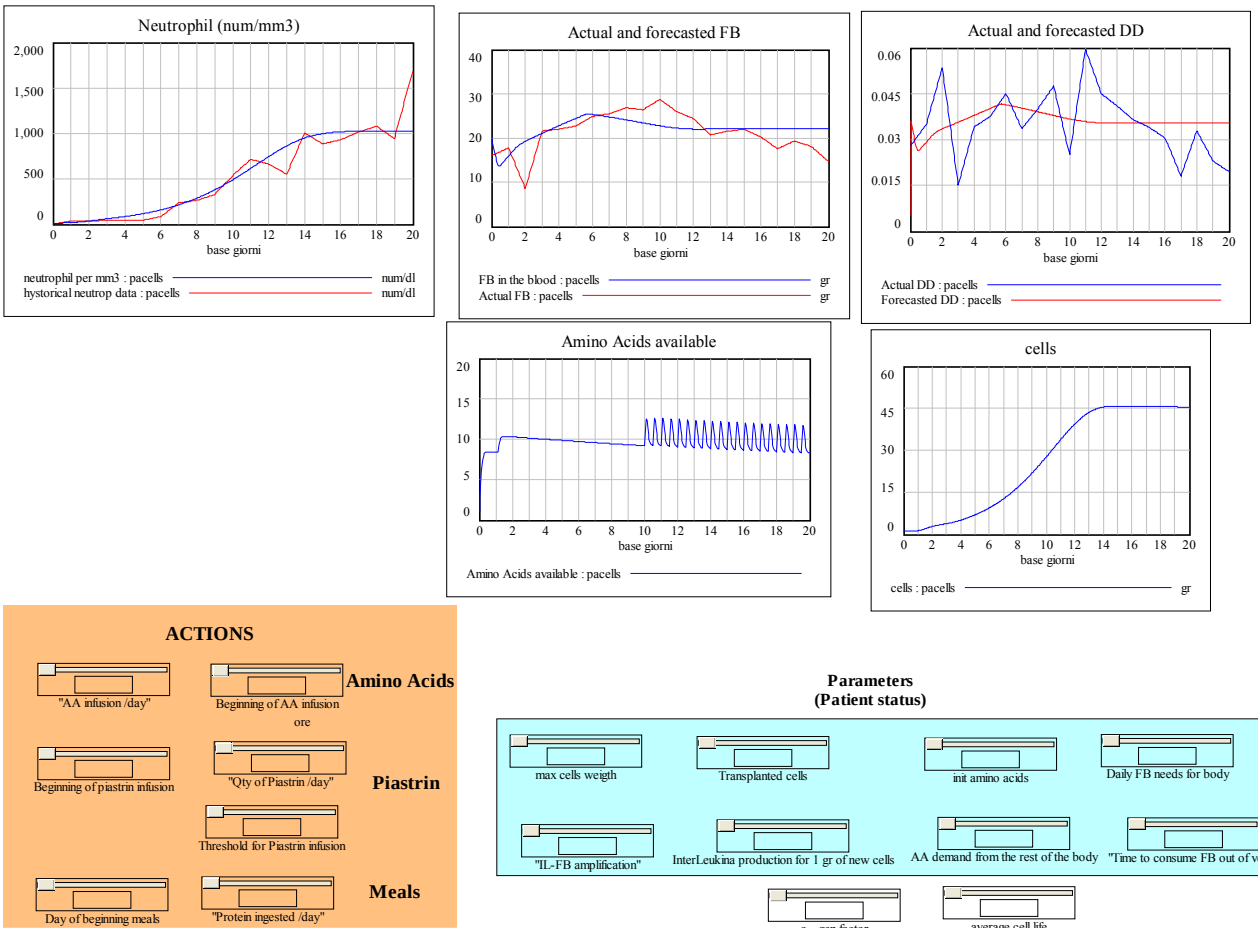

Fig. 4 A Control Panel