In-silico virtual prototyping multilevel modeling system for Cyborgs (CybSim) as a novel approach for current challenges in biosciences

Manuel Prado-Velasco

Department of Graphic Engineering and Multilevel Modeling in Bioengineering Group, University of Seville, Spain, mpradov@us.es

Abstract

There is a lack of Modeling and Simulation software systems in the bioscience arena that give both solutions compliant with current methodologies in drug discovery (pharmaceutical) and precision medicine (healthcare) fields, besides to support the addition of new biological mechanisms under a multilevel and multiformalism perspective, without penalize strongly the model sharing and reusing. A novel modeling and simulation software that tries to fill the previous gap has been designed (CybSim) and is presented in this work. CybSim is a platform for multilevel modeling of physiological - cybernetic systems, compliant but not limited to Physiologically based Pharmacokinetic and Pharmacodynamic (PBPK/PK/PD) methodologies. This capability is governed through the Physiological Scope setting value. The main physiological components are mechanistic. The underlying mechanisms may be changed during the model building thanks to the separation between mechanisms and physiological instances. This capability is based on a multi-layer design. A preliminary version of CybSim has been implemented with OpenModelica (v1.14.1). A PBPK semiphysiological model published previously has been built as a case study to demonstrate the feasibility of CybSim. The accuracy of CybSim was verified during preliminary development phases. The two pointed out capabilities of CybSim demanded an object-oriented and acausal equation-based modeling language, able to support classes’ redeclaration, connectors’ causality, inner/outer scoping control and packages organization. These features are not supported by other modern acausal equation-based modeling languages like the EcosimPro language.

Keywords: Cyborgs, Physiological modeling, PBPK, Mechanistic Modeling, acausal equation-based Modeling

1 Introduction

The field of modeling and simulation in biosciences is a mature domain that joins efforts from many areas, including biomedical engineering, mathematical biology, and pharmacology. Two big projects that started at the end of 1990’s and 2000’s may be cited as reference efforts in bioscience modeling. The Physiome initiative was presented in a report from the Commission of Bioengineering in Physiology to the International Union of Physiological Sciences (IUPS) council at the 32nd World Congress in Glasgow (UK) in 1993 (Hunter 2006, Box 1). The Virtual Physiology Human (VPH) project was initiated by the European Commission in 2007, after the publication of the Strategy for a European Physiome (STEP) (Hunter and Viceconti 2009). Both initiatives share as ultimate goal the research and development in computational models, data and tools for a better comprehension of human body under an integrated approach. Many projects defined under the umbrella of Physiome and VPH have pushed the development of a software framework for building mechanistic mathematical multiscale models from cellular to organ levels, with clinical, pharmacological and scientific applications.

A primary objective of VPH - Physiome initiatives was the establishment of model standards and repositories for which the well-known Extensible Markup Language (XML) was selected as a basis approach. Two relevant examples of modeling standards are CellML1 and SBML2. CellML was created for the modeling of cell-level dynamics, whereas SBML is a modeling language for networks of biological compounds (e.g. metabolic pathways) described using a systemic approach (Systems Biology). SBML and CellML encode models, metadata, and data, which represent the cited spatial scales (levels) of the physiological system to be modelled, in a robust and accurate manner. However, CellML and SBML models must be linked during the numerical integration to simulate the full system. The modeling and simulation tool that perform this task is an Achilles’ heel in this process, because it should be compliant with CellML and SBML and other standard modeling languages that represent the remaining physiological levels (tissues, organs, and living system). An interesting example is the Cardiac Physiome Project shown in Hunter and Viceconti (2009, Figure 6). In practice, the integrated framework must manage the interaction among models executed in different tools (Sauro et al. 2004). This approach is limited to interacting models that represent systems with weak coupling. On
the contrary the accuracy and stability of the full model is
degraded due to the jacobian deformation.

The increasing complexity of biosciences compels the
model standards to evolve. For instance, the SBML v3
standard tries to give response to the requirements of new
mathematical methods and physiological methodologies
(Keating et al. 2020). This is a second Achilles’ heel
of XML modeling languages, since mechanisms and ap-
proaches are restricted to fulfill those data formalisms and
levels. As a consequence, although the SBML, CellML and
others XML based modeling languages are designed to
build multiscale and modular models under a reusable
time, the two aforementioned process limitations dif-
cult the model reusing.

Due to the population requirements of the pharmacol-
ogy industry, the Non Linear Mixed Effect (NLME) mod-
els have guided the building of models in this field (Bon-
ate 2011). In addition, Pharmacokinetics (PK) and Phar-
macodynamics (PD) methodologies have defined during
decades the basis of the deterministic block of NLME
models due to their simplicity and success in the descrip-
tion of drug distribution and clinical responses. The PK
methodology has evolved to the Physiologically based
PK approach (PBPK) to consider anatomical and physi-
ofeological features and to improve the predictive ability of
NLME models, which is required to address with Drug-
Drug interactions (DDI) and special or vulnerable popu-
lations, for which clinical trials are not suitable (Jones,
Gardner, and Watson 2009). As a consequence, most
of the mature commercial software tools on pharmacol-
y industry are based on NLME with PK/PD (NON-
MEM (2021)), or PBPK/PD (SimCyp (2021), GastroPlus
(2021), Open-Systems-Pharmacology (2021)).

The gold standard in Population PK/PD modeling,
NONMEM, has driven a different approach to achieve the
model reusing and sharing in pharmacological sciences.
Despite NONMEM models are built through their ordi-
nary differential equations (ODEs), the wide diffusion of
NONMEM, R mathematical software⁵ and SBML, pro-
moted the development of a new model standard, Pharm-
ML, with the aim of promoting the model sharing. A
PK/PD model developed in PharmML is managed by a
compliant software tool that may also import SBML code and
convert the final model to NONMEM, R, SymCyp, and other
well-known modeling software tools (Bizzotto et al. 2017).

The pharmacology industry has pushed the acceptance and
and evolution of physiological models in the framework of
the Quantitative Systems Pharmacology (QSP), which is
an approach to translational medicine that combines com-
putational and experimental methods to the development
and use of molecules and biologic drugs at the beginning of
2000’s (Azer et al. 2021). Many advances in QSP
Modeling are supported by PBPK/PK/PD-based NLME
approaches with increasing efforts to facilitate the inclu-
sion of new knowledge discovering and modeling strateg-
ies from Physiome - VPH projects. However, PharmML
does not give a solution for the model sharing and reusing
requirement in QSP.

A different strategy to achieve the model reusing comes
from modeling languages based on equations’ formalisms
that do not depend on the algorithmic causality (Roa and
Prado 2006, Figure 8). This approach has proved its feasibil-
ity in the engineering field with EcosimPro lan-
guage (EL) (Empresarios Agrupados 2019) and Model-
cia (Fritzson 2015) as two cutting edge object-oriented
(00) and acausal equation-based modeling language refer-
ces. EL is a proprietary language implemented in the
EcosimPro software tool from Empresarios Agrupados In-
ternational (EAI), whereas Modelica is a freely available
language from the Modelica Association, implemented in
many open and proprietary software tools. This modeling
approach has been hardly applied in the biosciences arena
so far.

Physiolibrary is a library of specialized Modelica com-
ponents for the building of complex physiological models
(Mateják et al. 2014) that tries to give a solution for the
lack of adoption of Modelica in biosciences. It emerged
from the construction of a large model of human physi-
ology, Physiomedical, which in turn is an extended Model-
ica version of the integrative human physiological model
called HumMod (Hester et al. 2011). The recent study
of Ježek et al. (2017) describes a methodology for cre-
ating cardiovascular system models with different com-
plexity based on Physiolibrary with the main objective of
demonstrating the feasibility of a standardized platform
for model reusing. They show an interesting model that
considers the complex interactions between cardiac cir-
culation and arterial systems, founded on a hierarchy of
subsystems that takes advantage of the encapsulation and
acausal equation-based nature of Modelica. However, nei-
ther Physiolibrary (Mateják et al. 2014) nor the derived
cardiovascular system model (Ježek et al. 2017) offer a
modeling and simulation software tool oriented to the spe-
cific requirements and challenges in biosciences. For in-
stance, it is not an easy task to adapt any of them for the
solution of parameterized population PBPK models that
predict the distribution, therapeutic response and potential
interactions of a drug, which is a current standard problem
in the pharmacology area.

PhysPK is a software tool for modeling and simula-
tion in biosciences implemented with EL that was de-
signed to fill the gap between open and specialized tools in
PBPK/PK/PD that offers multilevel model reusing (Prado-
Velasco 2016). It was created thanks to an agreement be-
tween EAI and me (2015 - 2018) in which I (intellectual
owner) worked as team leader, designer and main devel-
oper. Several studies have shown the feasibility and ac-
curacy of PhysPK to develop population PK and PBPK
models, bioequivalence analysis, and even to generate pre-
dictive engines for precision medicine (Reig-Lopez et al.

⁵www.r-project.org/
2020; Gonzalez-Garcia et al. 2017; Prado-Velasco, Borobia, and Carcas-Sansuan 2020). However, the extension from PBPK to other modeling approaches is limited through the change of the input/output role in selected variables during the translation process, what induces numerical problems in non-desired flow-pressure transients (any PBPK model in PhysPK is a cardiovascular model). The impossibility to associate chemical names to the enumerated values and the difficulty to manage causal blocks are additional limitations.

In addition, neither Physiolab nor PhysPK has the capability to select the physiological mechanisms during the model building.

In summary, to the best of my knowledge, modeling and simulation software systems fail to provide a standardized framework for specialized bioscience areas like pharmacology (Maharao et al. 2020), precision medicine (Pollasek, Shakib, and A. Rostami-Hodjegian 2019; Darwich et al. 2017), toxicology (Paini et al. 2019; Bloomingdale et al. 2017), and regulatory decision (Rowland, Lesko, and Rostami-Hodjegian 2015; Shepard et al. 2015), with multilevel and evolutive model reusing and sharing capability.

The goal of this paper is to show a preliminary version of a novel multilevel modeling and simulation software for physiological - cybernetic systems (Cyborgs) based on Modelica and implemented with OpenModelica 1.14.1, called Cyborg Simulator (CybSim). It has been designed for the aforementioned biosciences areas with emphasis in model reusing. CybSim models may be built according to different modeling methodologies, including PBPK/P-K/PD, and the physiological mechanisms can be selected during the model building, what facilitates the model evolution. The diffusion of therapies where a machine is linked to the human body, in a temporal (hemodyalizer) or more permanent manner (artificial heart or insulin pump), is considered through the inclusion of a dedicated machines package. The current presence of computational control and logic in almost all therapy machines explains the selection of Cyborgs as system target. CybSim will be under open source license and available for download.

The work is divided in two stages: a brief presentation of the CybSim design (first) and a study case based on the semiphysiological PBPK model of Mangas-Sanjuan et al. (2018) to demonstrate the feasibility of CybSim (second). The model from Mangas-Sanjuan et al. (2018) was implemented in NONMEM and PhysPK (Reig-Lopez et al. 2020). The experience achieved in that study has facilitated a preliminary and succinct comparison of CybSim against PhysPK. The accuracy of the CybSim PBPK model was verified during the previous development stages.

It is noted that a detailed analysis of more complex physiological models exceeds the scope of this paper, what justifies the use of very single mechanisms in the study case.

2 Methods

The study is divided in two stages. The CybSim design is briefly presented in the first stage, which includes two steps.

1. General perspective of CybSim. It includes the package organization and some main setting properties.

2. Modeling strategy. The concept of multilayer design is explained and associated to the separation between mechanisms and physiological entities. Some design concepts related to the machine and signals packages are also presented in this context.

The second stage develops a study case based on a semiphysiological PBPK - based model. It comprises two steps.

1. Building of a semi-physiological model that includes intestinal lumen, gut, liver, a systemic plasma compartment (central) and a peripheral compartment. A solid form of parent drug (PD) is administered. This drug is metabolized in a principal metabolite (PM) and secondary metabolite (SM). The model is presented succinctly in the study case Section, although a detailed description is available (Mangas-Sanjuan et al. 2018; Reig-Lopez et al. 2020).

2. The CybSim model was executed under a periodic administration of the PD solid form, first with the original liver mechanisms of the reference model (Mangas-Sanjuan et al. 2018), and second after the modification of the molecular binding mechanism in the liver.

The study case tries to show the feasibility of CybSim to support the change of any underlying mechanism of a physiological instance during the model building. Other aspects of the CybSim modeling exceed the scope of the paper. The accuracy of the model was verified during the building process.

3 CybSim design

3.1 General properties and architecture

Some key issues of the CybSim design are presented in this Section. They referred to the preliminary version 0.2, implemented in OpenModelica 1.14.1. The Figure 1 shows the package’s organization of CybSim. These are grouped as follows:

- Specific units, main properties and simulation modes of CybSim: SUnits, Properties.

- Connectors and partial classes for the main interfaces: Interfaces.

- Packages that manage biological and physiological data: BioData.
In-silico virtual prototyping multilevel modeling system for Cyborgs (CybSim) as a novel approach for current challenges in biosciences

Figure 1. Packages' organization of CybSim v0.2.

- Packages that support the three CybSim layers: mechanisms, machines and physiological components, and data-driven models: Mechanisms, Machines, Physiology, Signals.

- Templates for model applications at different fields: ModelsInterfaces.

- Auxiliary packages with mathematical methods and Optimization procedures: Math and Optimization.

The main features of CybSim are summarized as follows:

- Physiological multiscope. Physiological subsystems in CybSim may be built according to different modeling methodologies, and thus with different scopes. CybSim v0.2 includes PBPK, LPhys and DPhys. A LPhys (general lumped parameter) model is a generalization of a PBPK model where the blood flow rates result from the cardiovascular circulatory system dynamics. Many times a PBPK, where blood flow rates are set by the modeler, is the right choice for a QSP study, and it has lesser computational requirements. The PBPK physiological scope in CybSim is compliant with the PBPK/PK/PD approach addressed in the Introduction. The DPhys refers to a spatially distributed modeling (not yet mature in CybSim). The Listing 1 shows the PhysScopeType enumeration type, which controls this feature and it is defined in the properties package.

- Free definition of chemical compounds for models. Chemicals compounds are addressed through a enumeration type (chemicals). The chemicalsDummy type (see Listing 1) is assigned as initial chemicals type, which must be redeclared in the final model, as shown in Listing 9. Several key entities in CybSim are defined in a vectorial (array) mode with chemicals as dimension. The physicochemical properties of the chemical compounds may be defined both manually or from a chemical database.

- Chemical volumes. CybSim v0.2 considers three modes of computing the chemicals volume in solution, governed by the ChemicalVolumeType enumeration (see Listing 1). The NoVolume mode (zero volume) is commonly used for small molecules, whereas the SmallAsSolvent considers the true dissolved density of large molecules. This feature requires the discrimination between no large and large molecules. A second enumeration type, mcrChemicals, which must be redeclared in the final model, defines the large molecules, whereas chemicals includes all of them.

- Multilevel modeling. Multilevel is addressed as synonymous of spatial multiscale. This feature is achieved by means of the aggregation (connection) of physiological components at each level, as shown in the conceptual structure of Figure 2 for the physiology subsystem. The same feature is available for the machines and signals subsystem. This is directly derived from the OO and acausal equation-based property of Modelica and it is available both in textual and graphical model according to the Modelica specification 3.4 (Modelica Association 2017) (MSLv35 has been recently delivered). A key issue here is that the granularity level of different tissues may be different.

- Multilayer architecture. The physical mechanisms that govern the dynamics of any physiological entity are selected during the model building. CybSim achieves this feature through the redeclaration of the inherited mechanisms following the methodology that is explained in the following Section. Conceptually, the physiological layer is defined as a set of physical and mechanistic components that may be defined at different scales (levels) as shown in Figure 2. The mechanism layer is not a physical but an organized set of abstract (partial) components. Although it is not presented in the Figure 2 for the sake of clarity, Machines and Signals may have a multilevel definition and they pertain to Machine and Signals layers, respectively, but they have not a sepa-
rated mechanistic layer. The fast evolution of the knowledge discovery in the biology field is the reason for this design criterion.

- Genes and Systems Biology. CybSim include metabolic networks based on a Systems Biology approach. This type of metabolic network is defined in the Mechanisms layer using a set of network components compliant with the Systems Biology Graphical Notation (SBGN), which requires the third enumeration type, genes. It is declared initially as the genesDummy type (see Listing 1), which must be redeclared in the final model if Systems Biology Metabolic Networks are used.

- Machines layer. A full machine-physiological system model is considered a cyborg model in CybSim. This generalization of a machine as a cybernetic component is based on the fact that near all machines designed for therapy, support or function enhancement include some type of automatic control systems.

- Signals layer. Data-driven or functional models (Roa and Prado 2006) are designed as Modelica blocks and organized in a package. PD standard models or PK metrics (e.g. Area Under the Curve, AUC) are included here.

A detailed description of the implementation of these features exceed the scope of this paper. However, some relevant issues are clarified in the following paragraphs.

Listing 1. Chemicals definitions and main CybSim scopes

```markdown
type chemicalsDummy = enumeration(
dummyCmp
) "Dummy Chemicals";
// ...
/* Mechanisms configurations */
type PhysScopeType = enumeration {
PBPK "Physiological -based Pharmacokinetics with parametric body Temperature",
LPhys "Lumped Physiology",
DPhys "Distributed Physiology"
} "Types of physiological approximations"

type chemicalVolumeType = enumeration {
NoVolume "First one is the most simple mode",
AllAsSolvent "All compounds with the same specific volume that solvent",
SmallAsSolvent "Non-small chemicals consider different volume"
} "Volumes of chemicals compared with solvent";
```

The physiological connectors depend on the scope type as expected, since blood flow rates are user defined with the PBPK scope, whereas they are solved according cardiovascular and circulatory system with the LPhys scope.

The Listing 2 shows the definition of a PBPK input connector as a causal connector that defines chemical and solvent flow rates, and the definition of a LPhys blood connector as an acausal connector that include chemical concentrations as Stream variables to address the reversion of blood flow rates. The later occurs for example in the arterio-venous fistula that connects a dialyzer with a patient. As a consequence, a PBPK model connected to a dialyzer cannot describe some operating conditions that occur with a clotted fistula in some patients. This is not a limitation of CybSim, but of the PBPK modeling approach.

The class bloodConnDummy allows controlling the types of Physiological connectors that may be rede-
clared in some multiscope CybSim components using the constrainedBy Modelica keyword.

Listing 2. Blood connectors

within CybSim;
encapsulated package Interfaces
// ...
partial connector bloodConnDummy
end bloodConnDummy;
partial connector inputDrag
replaceable type chemicals = Properties.chemicalsDummy "Chemicals in Cyborg";
input SI.VolumeFlowRate bvf (displayUnit = "l/min") "Causal input volume flow rate";
input SI.MassConcentration c[chemicals](each displayUnit = "ug/l") "Concentration of chemicals";
end inputDrag;
// ...
/* PBPK connectors */
connector inputBlood "Blood input connecto with PBPK scope"
extends bloodConnDummy;
extends inputDrag;
end inputBlood;
// ...
/* LPhys connectors */
connector bloodPort
extends bloodConnDummy;
replaceable type chemicals = Properties.chemicalsDummy "Chemicals in Cyborg";
SI.Pressure ps "static pressure at connection point";
flow SI.VolumeFlowRate vf "inlet (>0) volume flow rate at connection port";
stream SI.MassConcentration c[chemicals](each displayUnit = "ug/l") "Concentration of chemicals";
end bloodPort;

A key declaration in Listing 2 is the replaceable type chemicals that is defined by a shorthand inheritance from the chemicalsDummy enumerated declared in the Listing 1. The type chemicals must be defined in the user’s PBPK model according to the chemicals compounds required by the modeler. This feature is not available in other acusal languages as EcosimPro language (EL) (Empresarios Agrupados 2019).

A machine component may connected to any physiological subsystem, and therefore it must adapt their connectors and behaviour to the selected physiological scope. This is achieved thanks to redeclare the machine model and connectors as a function of the physiological scope, during model building. The Listing 3 presents the technique used for the pharmaceutical drug form machine shown in Figure 3. In this case, the DrugConn connector and the DrugFormScope model that describes the drugForm behaviour may be redeclare, providing that connectors derive from bryConnDummy and machine models de-
The FLT classes structure presented in Figure 5 was designed to allow the selection of the underlying physical mechanisms during model building. The eligible mechanisms for a PBPK component are organized according to functional types in the sgnGeneralPBPK partial component (Figure 5). A similar partial component, called generalPBPK is used if the physiological component does not require connectors to the Signals layer.

In agreement with the Figure 5, the fundamental structure of a PBPK FLT component appears in Listing 4. In this example, the mechanisms are inherited through the partial class generalPBPK. The equations section that appears commented in Listing 4 include very basic equations that complete the definition of the flt connector variables.

Listing 4. Flow limited tissue code structure

```model flt "flow limited tissue"
extends Icons.CausalEntity;
extends VFP.generalPBPK;
parameter Integer nBin = 1 "number input
blood perfusion volume flow rate"
annotation(Dialog(connectorSizing = true) , Evaluate = true);
parameter Integer nBout = 1 "number of
output blood perfusion volume flow
rate vias"
annotation(Dialog(connectorSizing = true), Evaluate = true);
Interfaces.inputBlood[nBin] inVBlood(
  redeclare type chemicals = chemicals)
"Input blood perfusions"
Interfaces.outputBlood[nBout] outVBlood(
  redeclare type chemicals = chemicals)
"Output blood perfusions"
equation
  /* Basic Equations related to systemic
  behaviour - connections */
  // ...
end flt;
```

As pointed in Figure 5, the physiological dynamics defined by the partial class sgnGeneralPBPK (and generalPBPK) is obtained through the inheritance of mechanisms organized by functional types. The code structure of generalPBPK partial class is shown in Listing 5. In opposition to sgnGeneralPBPK that inherits volRegionSgnPBPK, the component sgnGeneralPBPK inherits directly innerWholeVarsPBPK since it has not conditional interfaces to the Signals layer (see right column of Figure 5). A deeper description of this component exceeds the scope of the paper.

Listing 5. general PBPK dynamics of a volumetric region

```partial model generalPBPK "General whole
mechanism for volumetric regions with
PBPK scope"
extends Interfaces.innerWholeVarsPBPK;
extends PP.MassBalance.MasterPBPK;
extends PP.Elimination.MasterPBPK;
extends PP.PhaseDistribution.MasterPBPK;
extends PP.ChemicalActivity.MasterPBPK;
extends PP.MolecularBinding.MasterPBPK;
equation
  // Implicit relationship among
  variables
```

The FLT classes structure presented in Figure 5 was designed to allow the selection of the underlying physical mechanisms during model building. The eligible mechanisms for a PBPK component are organized according to functional types in the sgnGeneralPBPK partial component (Figure 5). A similar partial component, called generalPBPK is used if the physiological component does not require connectors to the Signals layer.

In agreement with the Figure 5, the fundamental structure of a PBPK FLT component appears in Listing 4. In this example, the mechanisms are inherited through the partial class generalPBPK. The equations section that appears commented in Listing 4 include very basic equations that complete the definition of the flt connector variables.

Listing 4. Flow limited tissue code structure

```model flt "flow limited tissue"
extends Icons.CausalEntity;
extends VFP.generalPBPK;
parameter Integer nBin = 1 "number input
blood perfusion volume flow rate"
annotation(Dialog(connectorSizing = true) , Evaluate = true);
parameter Integer nBout = 1 "number of
output blood perfusion volume flow
rate vias"
annotation(Dialog(connectorSizing = true), Evaluate = true);
Interfaces.inputBlood[nBin] inVBlood(
  redeclare type chemicals = chemicals)
"Input blood perfusions"
Interfaces.outputBlood[nBout] outVBlood(
  redeclare type chemicals = chemicals)
"Output blood perfusions"
equation
  /* Basic Equations related to systemic
  behaviour - connections */
  // ...
end flt;
```

As pointed in Figure 5, the physiological dynamics defined by the partial class sgnGeneralPBPK (and generalPBPK) is obtained through the inheritance of mechanisms organized by functional types. The code structure of generalPBPK partial class is shown in Listing 5. In opposition to sgnGeneralPBPK that inherits volRegionSgnPBPK, the component sgnGeneralPBPK inherits directly innerWholeVarsPBPK since it has not conditional interfaces to the Signals layer (see right column of Figure 5). A deeper description of this component exceeds the scope of the paper.

Listing 5. general PBPK dynamics of a volumetric region

```partial model generalPBPK "General whole
mechanism for volumetric regions with
PBPK scope"
extends Interfaces.innerWholeVarsPBPK;
extends PP.MassBalance.MasterPBPK;
extends PP.Elimination.MasterPBPK;
extends PP.PhaseDistribution.MasterPBPK;
extends PP.ChemicalActivity.MasterPBPK;
extends PP.MolecularBinding.MasterPBPK;
equation
  // Implicit relationship among
  variables
```
\[ e_i = c_i \cdot c_{u,i}, \tag{2} \]

in which the term \( c_{u,i} \) is the intrinsic clearance of chemical \( i \) in

\[ e_i = \left( K_{e,i} + \frac{V_{em,i}}{C_{i}} \right) \cdot C_i, \tag{1} \]

in which the concentration \( C_i \) is equal to the unbound significant phase (tissue) concentration \( c_u \) if the mechanism’s parameter significantPhase is true, and to the unbound non-significant phase (vesous) concentration, \( c_{em} \) otherwise. In homogeneous volumetric regions both concentrations are the same.

Assuming that significantPhase is true, the Equation 1 may also be written as follows:

\[ e_i = C_{i} \cdot c_{u,i}, \tag{2} \]

The eligible physical mechanism (see Figure 5) is called MasterPBPK and it is organized in packages according to the behaviour type. The Listing 6 shows the code of MasterPBPK for the elimination behaviour. The class dummyPBPK is defined to control the eligible elimination mechanisms.

Listing 6. Master model of the elimination dynamics in a volumetric region

```plaintext
partial model MasterPBPK "Master model for Elimination mechanisms with PBPK scope"
extends innerParamsPBPK;
replaceable class EliminationType =
NullPBPK constrainedby dummyPBPK
annotation(choicesAllMatching = true);
EliminationType mechElimination(
  //... replaceable class
  //... redeclare type
  //... redeclare type
end MasterPBPK;
```

The default elimination mechanism, Elimination.NullPBPK is the null elimination mechanism. Any volumetric region with this elimination mechanisms does not eliminate chemical compounds. The linear plus Michaelis - Menten (saturable) elimination is a well-known mechanism that describes this type of behaviour in Pharmacokinetics. The removal of any chemical \( i \) according this one is as follows:

Figure 5. Simplified classes structure of the sgnBryFlt component (left blocks) in the Physiology layer, which inherits the mechanisms sgnGeneralPBPK where it is defined the component dynamics. The sgnGeneralPBPK partial component is defined through a set of replaceable classes. Each one describes a type of eligible behaviour (metabolism, elimination, binding, etc.) related to a spatial volumetric region (partial processes, middle blocks). They are defined through equations that govern the variables of the spatial region, declared as inner variables in the innerWholeVarsPBPK partial component. Partial mechanisms work with the associated outer variables. The sgnGeneralPBPK full mechanism inherits volRegionSgnPBPK (right block) that in turn inherits innerWholeVarsPBPK and declare some conditional structures that simplifies the connection of metrics and other functional models of the Signals layer to the sgnBryFlt component.
the region and phase considered, given as:
\[ Cl_i = K_{e,i} + \frac{V_{em,i}}{K_{em,i} + c_{u,i}} \]  

(3)

The variables \( c_{u,i} \) must be substituted by \( c_{nu,i} \) in equations (2) and (3) if the elimination occurs in the non-significant phase. A detailed description of the tissue intrinsic clearance for a well-stirred region is shown in (Pang and Malcolm Rowland 1977). The elimination mechanism related to Equation 2 is implemented in Listing 7.

### Listing 7. Saturable elimination dynamics in volumetric region

```model
partial model clearanceSat "Sat+Lin elimination in volumetric region"
  extends outerCommonVars;
  outer parameter SI.VolumeFlowRate Ke[chemicals] "Clearance of chemicals";
  outer parameter SI.VolumeFlowRate Kesolv "Clearance of solvent";
  outer parameter SI.MassConcentration Kem[chemicals] "Michaelis Menten constant";
  outer parameter SI.MassFlowRate Vem[chemicals] "Michaelis Menten velocity";
  equation
    if significantPhase then
      for i in chemicals loop
        e[i] = cu[i] * (Ke[i] + Vem[i] / (Kem[i] + cu[i]));
      end for;
    else
      for i in chemicals loop
        e[i] = cnu[i] * (Ke[i] + Vem[i] / (Kem[i] + cnu[i]));
      end if;
    end if;
end clearanceSat;
```

### Figure 6. Flow limited tissue (ft) PBPK component with blood ports (circles left-right), boundary port (rectangle below) and signal connections (triangles) sgnBryFlt.

### Listing 8. Declaration of a FLT instance (central) in a PBPK model

```model
CybSim.Physiology.PBPK.Basic.sgnBryFlt
  central(
    redefine type chemicals = chemicals,
    redefine class EliminationType = CybSim.Mechanisms.VolRegion.
      PartialProcesses.Elimination,
      clearanceSatPBPK, Ke = {0, 5.5e-6, 8.3 e-6})
```

## 4 Study case

Figure 7 shows the diagram of the model defined in the Methods Section. A detailed description of this one appears in (Mangas-Sanjuan et al. 2018), whereas the comparative analysis of the PhysPK vs NONMEM implementations may be seen in (Reig-Lopez et al. 2020). The building of this model was performed using the GUI of openModelica 1.14.1 (diagram view), although the selection of non-default mechanisms for the physiological components were completed in the Modelica code view of the model, because openModelica 1.14 does not include this graphical function.

The model has been parameterized for a low dose of PD (100 mg) administered with two intakes of 50 mg separated 12 h, for a high absorption rate constant in the gut (drug of class II of Biopharmaceutics Classification System), saturable metabolism in gut and liver, and a reference value of the dissolution rate constant (quality level).

The parameter values agree with those used in the first case of (Mangas-Sanjuan et al. 2018, Fig. 2), excepting the division of the 100 mg PD dose in two separated 50 mg PD doses, which is applied now in agreement with the second step pointed in Methods to analyze a more complex scenario, after validating the accuracy of the model against the results of (Mangas-Sanjuan et al. 2018) and (Reig-Lopez et al. 2020).

The final objective of this case study is to demonstrate the feasibility of CybSim to select and change the underlying mechanisms of the model physiological instances during the building process. With this goal, after simulating the model with the liver mechanisms defined in (Mangas-Sanjuan et al. 2018) (first simulation), a linear molecular binding mechanism is added to the liver and a new simulation is executed (second simulation).

The Listing 9 shows the structure of the Modelica code of the semiphysiological model, including the definition of the Liver tissue. The chemicals enumeration is set through the three required chemicals compounds, PD, PM, and SM. The mechanism associated with the chemical activity in the liver is declared as a saturable metabolism (MichaelisMentenPBPK), during the initial model building. The boolean inputs biInst...
Figure 7. Semiphysiological PBPK model published in (Mangas-Sanjuan et al. 2018; Reig-Lopez et al. 2020) that includes a drug form (drF) from the machine layer, and two PK metric blocks (AUC and saUC) from the signals layer.

and biPlan define a true planned and false instantaneous administration mode for the drug form, as wished.

The execution of the model for a temporal window of 24 hours gives the plasmatic concentrations of PD, PM and SM under the defined scenario (first simulation).

Listing 9. Main code structure and liver definition of semiphysiological model

```plaintext
model drF2cmpLivGut
  extends CybSim.ModelsInterfaces.Interfaces.pbpkStr(redeclare type chemicals = chemicals);
  import CybSim.Properties;
  import PQ = CybSim.BioData.PhysicalChemical;
  /* Chemical definitions */
  type chemicals = enumeration (PD "Parent drug", PM "Primary metabolite", SM "Secondary metabolite");
  CybSim.Physiology.PBPK.Basic.flt Liver(
    redeclare type chemicals = chemicals,
    redeclare class ChemicalActivityType = CybSim.Mechanisms.VolRegion.
      PartialProcesses.ChemicalActivity.
        MichaelisMentenPBPK,
        Kmm = {0.1, 0.01, 0.01}, Vmm = {0, 1e-9, 1e-9}, reactantsm={
          chemicals.PD,chemicals.PD,chemicals.PD,chemicals.PD },
        Vtis0 = 0.003, nBin = 2, nBout = 1)
  //...
  equation
  //...
  drF.biInst = false;
  drF.biPlan = true;
  end drF2cmpLivGut;
```

The Listing 10 shows how a linear molecular binding is applied to substitute the default (null) molecular binding of the liver instance of the semiphysiological model.

The linear molecular binding mechanism modifies the amount of free drug according to the unbound fraction drug values $f_u$. These fractions are defined in the liver declaration as 5% (0.05) for the PD, and 100% (1) for the metabolites. That is, the PD is the unique compound that is bound to a macromolecule, in such a way that only 5% is free. This is a very common situation in physiological models.

Listing 10. Definition of the FLT liver instance with linear binding

```plaintext
CybSim.Physiology.PBPK.Basic.flt Liver(
  redeclare type chemicals = chemicals,
  redeclare class MolecularBindingType = CybSim.Mechanisms.VolRegion.
    PartialProcesses.MolecularBinding.
      linearPBPK, fu = {0.05, 1, 1}, ...)
```

The execution of the model for the same temporal window of 24 hours gives the plasmatic concentrations of PD, PM and SM in the second simulation.

5 Simulation results and discussion

Figure 8 shows the plasmatic (central) concentrations of the parent drug and their metabolites during the first 24 hours, starting with the first PD dose (50 mg).

The temporal distribution of the chemical compound was accurately validated both for an unique dose of PD equal to 100 mg (Mangas-Sanjuan et al. 2018, Fig. 2) and for two sequential doses of PD equal to 50 mg. A detailed analysis of this testing phase exceeds the scope of this paper that is focused to demonstrate the feasibility of the CybSim design to support the mechanisms - physiology architecture.
The Figure 9 shows the plasmatic concentrations of PD and their metabolites during the first 24 hours for the second simulation. As expected, the PD concentration increases, whereas PM and SM concentrations decrease, with respect to the first simulation, due to the reduction of liver metabolism because of the smaller amount of free PD.

The Area Under the Curve (AUC) blocks, which are connected to the concentrations (all the chemicals) inside the central compartment (AUC block) and to the total mass flow rate from central to peripheral compartment (sAUC or single AUC), are not presented here. However, they were used to evaluate the parent drug and metabolites AUC in the target central compartment and to calculate the total net amount of mass between central and peripheral compartment during the first 24 hours. They were used also to demonstrate the capabilities of CybSim related to the Signals layer.

Although a detailed comparison between CybSim and PhysPK exceeds the scope of this paper, CybSim overcomes several important limitations of PhysPK due to the lack of classes’ redeclaration, inner/outer structures, packages organization, and causality of connectors. These language characteristics are the basis of the mechanisms - physiology multilayer, the multiscope feature, the free definition of chemical compounds inside the physiological instances, and many of the capabilities of the signals layer.

Summary

This study has presented a novel Modeling and Simulation software system for the biosciences field, CybSim, that gives a framework compliant with current methodologies and specific solutions required in particular areas like pharmacology, precision medicine and toxicology. CybSim tries to overpass some detected lacks related to model reusing and sharing in current Modeling and Simulation software systems.

The outcomes demonstrate the feasibility of CybSim to facilitate the choice of the mechanisms underlying the physiological entities in the process of model building. To the best of my knowledge, this is the first biosimulation system that fulfills that feature at the same time that offers modeling multiscope, free definition of chemicals, multilevel modeling, metabolic networks based on the systems biology approach, machines integration and support for data-drive modeling.

Future works will be developed to evaluate those features, and to complete the implementation of other packages related to optimization (applied to population estimation and dosage personalization), and biodata (algorithms for in-vitro in-vivo correlation, allometric scaling, and methods for computation of physiological properties).

This is a first paper concerning the preliminary version 0.2 of the CybSim biosimulation system. More advances will be performed and published shortly, including the evolution to the available openModelica 1.18 that should give better solutions to some relevant planned features. CybSim will be deployed with availability for download under open source license, after reaching the required minimal functionality.

References

In-silico virtual prototyping multilevel modeling system for Cyborgs (CybSim) as a novel approach for current challenges in biosciences