Proceedings of the 29th Annual International
Conference of the IEEE EMBS
Cité Internationale, Lyon, France
August 23-26, 2007.

A Generic Representation Format of Physiological Experimental Protocols for Computer Simulation using Ontology

Takao Shimayoshi, Akira Amano and Tetsuya Matsuda

Abstract— **For computer simulations of physiological experiments using physiological models, a machine-readable format of experimental protocols is effective. Here, we propose an XML-based language, PEPML (Physiological Experimental Protocol Markup Language). In the PEPML, conditions and procedures of an experimental protocol are procedurally described as a list of events, each of which consists of a condition for execution and an action to be executed. Since variables used in a protocol can be specified using an ontology, the protocol can be applied to various models without editing tasks. The PEPML allows both application of multiple protocols to a single model and application of a single protocol to multiple models. This feature facilitates the efficient simulation for verifications, comparisons and utilizations of physiological models.**

I. Introduction

Modelling of physiological cell functions is of importance for the understanding of cellular mechanisms, which in turn leads to advancement in biological research as well as in medical and pharmaceutical technologies. Many models of physiological cell functions are, despite being already published, constantly being improved and extended towards more precise and comprehensive ones. During the development and application of such mathematical models, computer simulations are necessary to analyze the behaviour. A machine-readable representation of models is valuable in making simulations straightforward. Indeed, several representation formats are developed including SBML (Systems Biology Markup Language)[1] and CellML[2]. CellML designed by the University of Auckland is a description format for mathematical models. In CellML, mathematical expressions of a model are described in separate components. Many models described in CellML were released on the CellML site (www.cellml.org).

Each simulation experiment with a physiological model is performed according to an experimental protocol in the same way as each wet experiment. For model validations and predictions using models, different experimental protocols need to be simulated with a single model. Furthermore, for a selection of various physiological models such as myocyte models in model applications, the same experimental protocol needs to be used for each model to compare them. A machine-readable description of experimental protocols as well as models would advance the efficient application of multiple protocols to a single model as well as the use of one experimental protocol for different models.

Here, we propose a new generic representation format for experimental protocols, PEPML (Physiological Experimental Protocol Markup Language), which can describe protocols independent of models in conjunction with an ontology. An experimental protocol in PEPML can be applied to a physiological model described in a representation format such as CellML.

II. Proposed Methods

A. Representation of Experimental Protocols

The protocol of a physiological experiment includes experimental conditions of the environment such as the composition of the external solution and the temperature, and experimental procedures such as the periodic electrical stimuli and the temporary blockade of an ion channel. A specialised format based on XML (Extensible Markup Language) to represent experimental protocols, PEPML describes protocols separately from models. PEPML supports experimental protocols that can be represented through operations with model variables. A dosing condition can be expressed as a quantitative drug effect, e.g. the blocking percentages of the effected channels. Describable protocols include most physiological experiments in cellular electrophysiology and biomechanics.

An experimental protocol in a PEPML file is described by a *protocol* element. Conditions and procedures of the protocol are represented as a list of events. An event has a condition for the execution and an action composed of multiple instructions to be executed, and is declared with an *event* element, which contains a *condition* element and an *action* element. A condition is declared as a combination of elements representing relational operators and Boolean operators. An action is declared as a sequence of elements expressing assignment statements which can include arithmetic operations and mathematical functions. Fig. 1

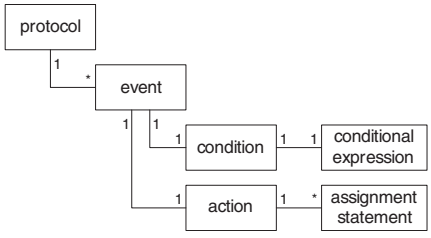


Fig. 1. Main structure of PEPML

This work was supported by the Leading Project for Biosimulation, Ministry of Education, Culture, Sports, Science and Technology, Japan.

T. Shimayoshi is with ASTEM Research Insutitute of Kyoto, Kyoto, 600–8813, Japan, and with Graduate School of Informatics, Kyoto University, Kyoto, 600-8501, Japan simayosi@astem.or.jp

A. Akira and T. Matsuda are with Graduate School of Informatics, Kyoto University, Kyoto, 600–8501, Japan

TABLE I

OPERATOR ELEMENTS OF PEPML

Relational operators
eq, gt, lt, ge, le
Boolean operators
and, or, not
Arithmetic operators
add, sub, mul, div, mod
Assignment operators
set_value, add_value, sub_value, mul_value, div_value

```
<?xml version="1.0"?>
<protocol>
  <event id="event1">
    <condition>
      <and>
        <ge><time /><literal value="10.0" /></ge>
        <eq><variable ref="x" /><variable ref="m" /></eq>
      </and>
    </condition>
    <action>
      <set_value>
        <variable ref="y" />
        <add>
          <literal value="1.0" />
          <sin><time /></sin>
        </add>
      </set_value>
      <mul_value>
        <variable ref="z" />
        <literal value="4.0" />
      </mul_value>
    </action>
  </event>
</protocol>
```

Fig. 2. A PEPML document

illustrates the main structure of PEPML as a UML (Unified Modelling Language) class diagram. Table I shows a list of operator elements supported by PEPML. In the condition and the action, operands of operators and functions are indicated with *variable* and *literal* elements. To demonstrate a PEPML document, a pseudo experimental protocol is shown in Fig. 2. During the execution of a protocol, if the conditions of multiple events are met at the same time, the actions of all events are processed in the order of description.

A *variable* element is referred to a variable in the target model by using the *ref* attribute. There can be done in two formats for the value of the *ref* attribute. One way is to specify a variable of a particular model directly:

```
<variable ref="compA.var1" />
```

A variable in a CellML file can be referred by a concatenation of the component name and the variable name with a single dot. The other way allows an independent reference to variables from model instances. In this case, the value of the *ref* attributes is an identifier of an ontology:

```
<variable ref="cmo:1234" />
```

To use this format, the target CellML files need to be associated with an ontology. Previously, we have developed a Cell Model Ontology[3] on components of physiological models, which is suitable for this application. To connect CellML files with the Cell Model Ontology, identifiers of the ontology instances are assigned to variables of CellML:

```
<component name="compA" ...>
```

```
<event id="tyrode">
  <condition>
    <always />
  </condition>
  <action>
    <set_value>
      <variable ref="cmo:ID_[Na]o" />
      <literal value="140" units="mM" />
    </set_value>
    <set_value>
      <variable ref="cmo:ID_[K]o" />
      <literal value="5.4" units="mM" />
    </set_value>
    <set_value>
      <variable ref="cmo:ID_[Ca]o" />
      <literal value="2.0" units="mM" />
    </set_value>
  </action>
</event>
```

Fig. 3. An experimental condition of external solution

```
<event id="default">
  <condition>
    <always />
  </condition>
  <action>
    <set_value>
      <variable ref="cmo:ID_stim_current" />
      <literal value="0.0" />
    </set_value>
  </action>
</event>
<event id="periodic stimuli">
  <condition>
    <and>
      <ge>
        <mod>
          <time /><literal value="400.0" units="ms" />
        </mod>
        <literal value="50.0" units="ms" />
      </ge>
      <lt>
        <mod>
          <time /><literal value="400.0" units="ms" />
        </mod>
        <literal value="52.0" units="ms" />
      </lt>
    </and>
  </condition>
  <action>
    <set_value>
      <variable ref="cmo:ID_stim_current" />
      <variable ref="cmo:ID_stim_amplitude" />
    </set_value>
  </action>
</event>
```

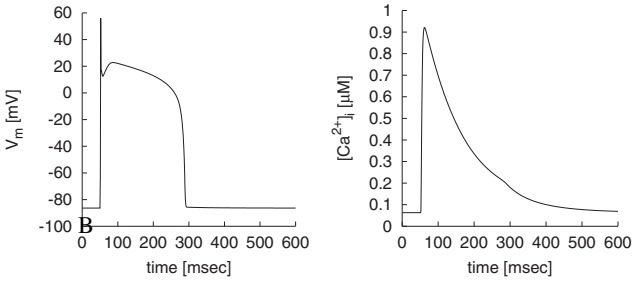
Fig. 4. An experimental procedure for periodic stimuli

```
<variable name="var1"
          cmo:id="1234" .../>
```

In addition, we also reported an estimation method to identify components and variables of CellML files with the Cell Model Ontology[3]. PEPML files which only use the ontology style for references can be applied broadly to different physiological models.

Fig. 3 shows an event which represents the experimental condition of external solution. In this event, three variables for external ion concentrations are always set to respective values throughout the experiment, i.e. these variables are kept constant. Fig. 4 demonstrates events of an experimental procedure for periodic stimuli to a cell. The first event sets the default value of the stimulation current to zero. Then, if

A



B

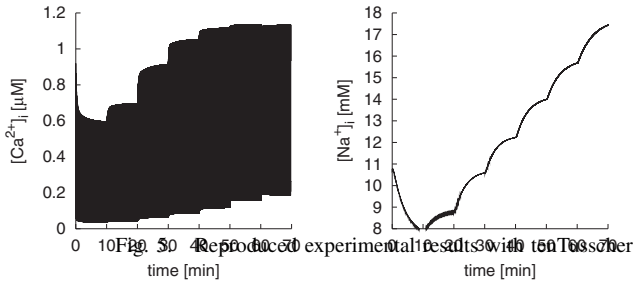


Fig. 5. Reproduced experimental results with tenTusscher model

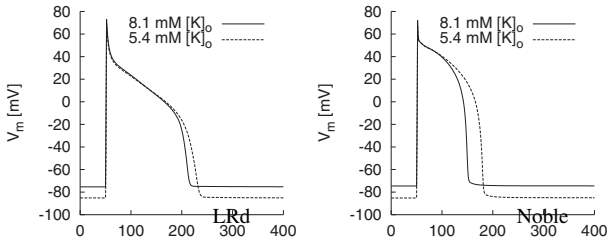


Fig. 6. Action potential under a hyperkalemic condition

the condition of the second event is met, i.e. periodically, the stimulation current is set to a specified value.

B. Simulation of Experimental Protocols

A computational physiological model is generally formulated as a set of temporal ordinary differential equations or temporal differential algebraic equations. In numerical simulation, a physiological model defines a target system of equations, and an experimental protocol is used as a boundary condition for these equations. To demonstrate how PEPML files can be processed in simulations, a simulator is implemented as a reference software, which performs a physiological experiment by applying an experimental protocol in PEPML to a cell model in CellML.

The software executes simulations using the following steps. First, equations in a target CellML file are converted to program code. Next, the calculation procedure of equations is determined by analyzing dependencies between equations, using an experimental protocol in a PEPML file as the boundary condition. Then, differential equations are explicitly solved in a stepwise fashion while sequentially checking and executing events of the protocol in each step.

III. Experiments

Simulation experiments were performed using PEPML and the implemented simulator. In these experiments, public CellML files are used with minor corrections and modifications, i.e. initial values of the variables are changed to the original values of the models and interleaved stimulation protocols are removed.

Fig. 5 shows the results for the experiments using the human ventricular cell model by tenTusscher et al[4], to reproduce the simulation experiments in their publication. An action potential and a calcium transient are shown in Fig. 5A. In these simulations, constant external ion concentrations, 5.4 mM $[K^+]_o$, 2 mM $[Ca^{2+}]_o$, and 140 mM $[Na^+]_o$ were employed for the "experimental conditions" and 1 Hz pacing for the "experimental procedure". Fig. 5B shows the changes in the intracellular calcium and sodium levels when the pacing frequency is changed from 0.25 to 0.5Hz and then in 0.5Hz steps from 0.5 to 3Hz with maintaining each frequency for 10 minutes. By applying these two PEPML files containing different protocols to the same CellML file, both experiments could successfully be reproduced almost matching the published[4].

Fig. 6 shows a comparison of action potentials of two guinea-pig ventricular cell models, the LRd model[5] and the Noble model[6] under a high extracellular potassium concentration, 8.1 mM $[K^+]_o$ and 1Hz pacing. These experiments were performed by applying the same PEPML file to these different CellML files. An increase of the $[K^+]_o$ from 5.4 to 8.1 mM results in APD_{90} shortening by about 11% and 30% for the LRd model and the Noble model respectively. As reported by Wan et al.[7], the percentage of APD_{90} shortening for sub-endocardial, mid-myocardial and sub-epicardial myocytes are 13%, 17% and 28%, respectively. The results for the LRd model and Noble model is similar to sub-endocardial and sub-epicardial cells.

IV. Discussion

The generic representation of experimental protocols is an effective method for computer simulation of physiological models. Machine-readable descriptions of protocols make experiments efficiently reproducible. In addition, separate declarations of protocols from models allow application of different protocols to each model, and a generic definition of a protocol facilitates the simulation of multiple models with the same protocol.

The proposed PEPML allows appropriate representations of generic experimental protocols. In fact, CellML, which is a representation format for physiological models, can be used to describe experimental protocols through declaration of mathematical expressions, ¡and allows separated descriptions of protocols from models by using the 'import' feature. However, an experimental protocol is procedures considered by experimentalists. In addition, it is difficult for some protocols to convert them into mathematical descriptions, e.g. doubling of a current value of a variable at a certain

time. Therefore, a procedural expression, which PEPML is based on, is more appropriate than a declarative one. Furthermore, difficult editing tasks are required to make connections between variables of a protocol CellML file and those of a model, which is an unavoidable step if an experimental protocol written in CellML is applied to a CellML model. In contrast, PEPML allows applications of protocols without any editing tasks by employing ontology. Since the representation of models and that of experimental protocols have different characteristics, descriptions using specialised formats for respective representations are efficient and intuitive for developers and users of models.

As shown here, each simulation can be executed by just describing the protocol in PEPML and applying it to the target CellML file. This fact proves that an independent description of experimental protocols by using PEPML makes simulations for model validation and comparison easier. Furthermore, a set of PEPML files can be a test case for different models. These features of PEPML facilitate model developments, curations and applications.

V. Conclusion

This paper reports PEPML, a generic representation format for physiological experimental protocols. PEPML is intuitive in the description of protocols because of its procedural format. A protocol can be described independently of physiological models by using our proposed ontology, Cell Model Ontology[3]. PEPML allows an efficient simulation of various experimental protocols with different physiological models. Although applications for models described in CellML are mainly discussed in this paper, PEPML could be applied to other formats including SBML. As discussed in MIRIAM[8], reproducibility of published models is fairly important. Distribution of models with experimental protocols in PEPML files can simplify reproduction of models. Furthermore, since a set of PEPML files can serve as a test case for models, PEPML will facilitate validation and verification of models.

VI. Acknowledgement

The authors wish to thank members of Cell/Biodynamics Simulation Project of Kyoto University for valuable comments during the study, and also thank Dr. Yu Shimizu and Dr. Natalie Schneider for improving the English of this paper.

References

[1] M. Hucka, A. Finney, *et al.*, "The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models," *Bioinformatics*, vol. 19, no. 4, pp. 524–531, Mar. 2003.

[2] A. A. Cuellar, C. M. Lloyd, P. F. Nielsen, D. P. Bullivant, D. P. Nickerson, and P. J. Hunter, "An overview of CellML 1.1, a biological model description language," *SIMULATION*, vol. 79, no. 12, pp. 740–747, Dec. 2003.

[3] T. Shimayoshi, K. Komurasaki, A. Amano, T. Iwashita, T. Matsuda, and M. Kanazawa, "A method to support cell physiological modelling using description language and ontology," *IPSJ Digital Courier*, vol. 2, pp. 726–735, Nov. 2006.

[4] K. H. ten Tusscher, D. Noble, P. J. Noble, and A. V. Panfilov, "A model for human ventricular tissue," *American Journal of Physiology. Heart and Circulatory Physiology*, vol. 286, no. 4, pp. 1573–1589, Apr. 2004.

[5] C.-H. Luo and Y. Rudy, "A dynamic model of the cardiac ventricular action potential. I. simulations of ionic currents and concentration changes," *Circulation Research*, vol. 74, no. 6, pp. 1071–1096, June 1994.

[6] D. Noble, A. Varghese, P. Kohl, and P. Noble, "Improved guinea-pig ventricular cell model incorporating a diadic space, I_{Kr} and I_{Ks}, and length- and tension-dependent processes." *Canadian Journal of Cardiology*, vol. 14, no. 1, pp. 123–134, Jan. 1998.

[7] X. Wan, S. M. Bryant, and H. George, "The effects of $[k^+]_o$ on regional differences in electrical characteristics of ventricular myocytes in guinea-pig," *Experimental Physiology*, vol. 85, no. 6, pp. 769–774, Nov. 2000.

[8] N. Le Novère1, A. Finney, M. Hucka, U. S. Bhalla, F. Campagne, J. Collado-Vides, E. J. Crampin, M. Halstead, E. Klipp, P. Mendes, P. Nielsen, H. Sauro, B. Shapiro, J. L. Snoep, H. D. Spence, and B. L. Wanner, "Minimum information requested in the annotation of biochemical models (MIRIAM)," *Nature Biotechnology*, vol. 23, no. 12, pp. 1509–1515, Dec. 2005.