BioUML – универсальный язык для визуального моделирования молекулярно-биологических систем

Ф.А. Колпаков

Конструкторско-Технологический Институт Вычислительной Техники СО РАН, г. Новосибирск; Biosoft.Ru.

fedor@biosoft.ru

Введение

В данной работе нами рассматривается подход для создания универсального графического языка для формализованного описания структуры и функции молекулярно-биологических систем в виде диаграмм, а так же моделирования их динамики. Под молекулярно-биологической системой (МБС) мы понимаем любую биологическую систему или ее часть, где ключевыми объектами рассмотрения являются клетки, гены, молекулы РНК, белок/белковые комплексы и их взаимодействия. Метаболические пути, генные сети и пути передачи сигнала являются частными случаями МБС.

На данном этапе разработки BioUML нами решаются следующие задачи:

1. Разработать универсальный поход для формализованного описания структуры и функционирования МБС в виде диаграмм разных типов (структурные диаграммы, диаграммы состояний, диаграммы классификаций и онтологии).
2. Обеспечить возможность реконструкции структуры МБС на основе взаимодополняющих данных полученных на разных видах организмов и в различных условиях. Обеспечить возможность выделения из такого обобщенного описания такой подструктуры, которая специфична для заданного вида организмов, типа клеток, или других условий.
3. Обеспечить возможность легкой интеграции существующих баз данных и способов представления структуры МБС в этих базах данных в среду BioUML.
4. Разработать интерфейс для построения математических моделей МБС на основе их графических диаграмм (с каждым ребром графа ассоциируется набор дифференциальных уравнений) и последующей автоматической генерацией моделей на языке MATLAB для их численного решения и анализа.

Ниже мы рассмотрим структуру среды BioUML. На примере базы данных GeneNet продемонстрируем процесс интеграции различных баз данных в среду BioUML в виде

отдельных модулей. На простом примере из фармокинетики рассмотрим моделирование МБС с использованием BioUML и MATLAB.

Среда BioUML

Среда BioUML (BioUML framework) является компьютерной системой на языке Java и состоит из следующих частей:

- *мета модель* – определяет уровень абстракции для описания структуры любой диаграммы в виде кластеризованных графов (см. ниже).
- *BioUML viewer* (рис. 3) – универсальная программа просмотра для графического представления диаграмм (МБС).
- *BioUML editor* – универсальный редактор диаграмм (на данный момент находится в процессе разработки).
- *BioUML search engine* – систему поиска компонентов МБС по базе данных и их взаимодействий друг с другом. Результаты поиска представляются в виде графа, вершинами которого являются найденные по запросу компоненты МБС, а ребра – их взаимодействия друг с другом. Система поиска находится в процессе разработки, по завершению она будет обеспечивать функциональность, сходную с системой поиска базы данных TRANSPATH (Schacherer et al., 2001), однако полученная диаграмма может быть расширена и отредактирована пользователем.
- *BioUML modeler* – используется для создания портретных моделей МБС в виде графических диаграмм. Более детально этот подход будет описан ниже.
- *Модули баз данных* – обеспечивают интеграцию различных баз данных в среду BioUML. Ниже мы рассмотрим данный подход на примере интеграции базы данных GeneNet.

BioUML мета модель

Мета модель (дословно модель модели) определяет уровень абстракции для описания структуры любой диаграммы в виде кластеризованного графа. Ее структура в виде диаграммы классов UML представелена на рисунке 1.

Все элементы диаграммы являются объектами класса `DiagramElement`, который определяет общие для них всех свойства:

- `kernel` – в общем случае это ссылка на конкретный объект из заданной базы данных. Таким образом мы можем ассоциировать с любым элементом диаграммы конкретную информацию из базы данных.

- `view` – графическое представление (как правило, векторная графика) элемента диаграммы. Оно создается `DiagramViewBuilder`.
- `title` – заголовок элемента на диаграмме. Обычно это идентификатор соответствующего объекта из базы данных, однако он может быть задан/модифицирован пользователем.
- `comment` – произвольный текстовый комментарий, ассоциированный с элементом диаграммы.
- `role` – роль (переменная или дифференциальное уравнение), которую играет данный элемент при построении математической модели МБС.

Далее элементы диаграммы делятся на ребра и вершины графа. Все ребра графа являются направленными (т.е. мы работаем с орграфами) и представляются в виде объектов классов `Edge`. Простые вершины графов представляются в виде объектов класса `Node`. Класс `Compartment` является наследником класса `Node` и используется для представления вершины графа, содержащей несколько других вершин. Такой подход позволяет упорядочить компоненты МБС (гены, белки, химические соединения и т.д.) по компартментам, а так же учесть иерархические взаимоотношения между компартментами (например, клетка включает цитоплазму, а цитоплазма включает ядро и клеточные органеллы).

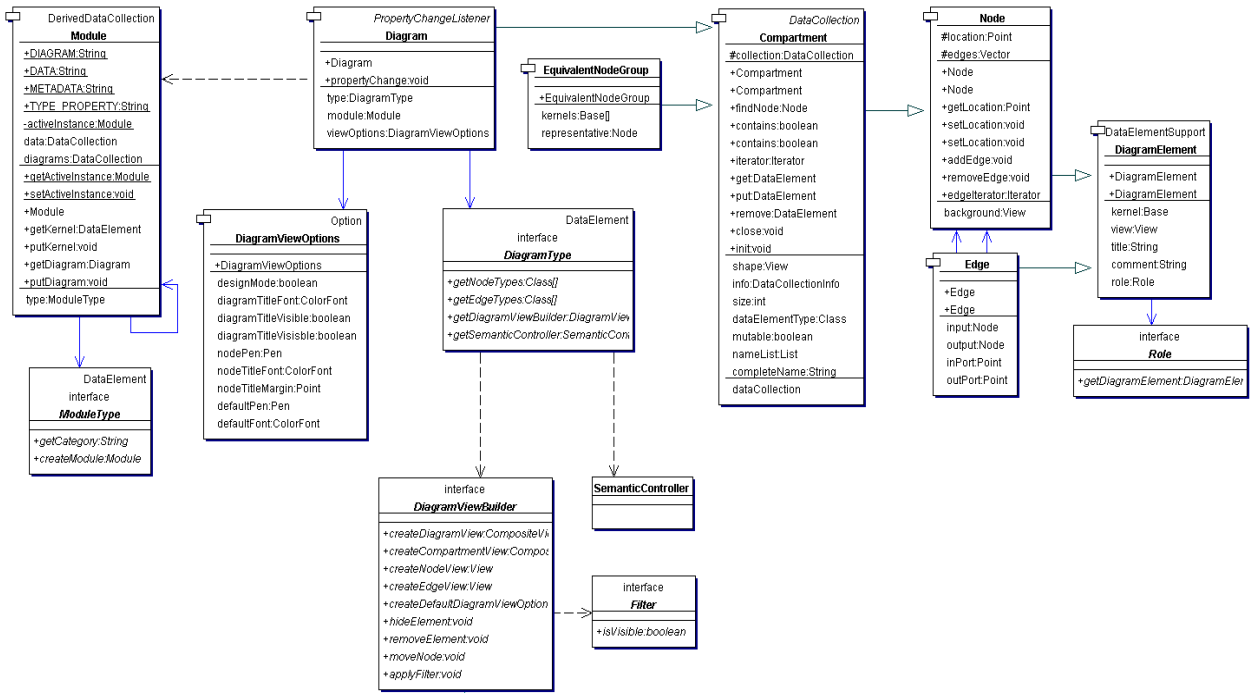

Рисунок. Диаграмма классов, определяющих BioUML мета модель.

Одной из задач перед BioUML является возможность реконструкции структуры МБС на основе взаимодополняющих данных полученных на разных видах организмов. В результате на диаграмме может быть несколько объектов (например, гомологичные гены альфа-глобинов разных видов организмов) эквивалентных в рамках диаграммы, обобщенной, например, по нескольким видам млекопитающих. Такие объекты объединяются в группу эквивалентности – специальный тип компартмента, представляемый в виде объекта класса `EquivalenceNodeGroup`. На диаграмме группа эквивалентных объектов может изображаться в виде одной картинки (например, ген альфа-глобина) или же в виде компартмента, содержащего все объекты группы.

Все диаграммы представляются в виде объектов одного и того же класса `Diagram`, а тип диаграммы задается через значение ее атрибута `type` – объект класса `DiagramType`. `DiagramType` задает способ графического представления диаграмм, используя класс `DiagramViewBuilder`, определяет какая информация из базы данных может быть использована в качестве вершин и ребер графов, а так же отвечает за семантическую правильность диаграмм, используя класса `SemanticController`.

Предложенный подход позволяет нам посредством классов `Node`, `Edge`, `Compartment` и `Diagram` задать универсальное представление структуры (модели) всех типов диаграмм, а информацию, специфичную для их внешнего вида вынести в специальные классы `DiagramType` и `DiagramViewBuilder`, расширяя которые, можно задавать новые типы диаграмм.

Концепция модуля

Чтобы обеспечить интеграцию различных баз данных в среду BioUML, мы вводим концепцию модуля. Если воспользоваться метафорой, то BioUML можно представить как операционную систему (например, Windows), а модули тогда будут являться отдельными программами (например, MS Word).

Как правило, модуль создается для отдельной базы данных и определяет способ представления информации из этой базы данных в виде объектов языка Java. Модуль также может содержать специфичные для этой базы данных типы диаграмм, представляемые в виде подклассов класса `DiagramType` и способы их графического отображения, задаваемые как расширения `DiagramViewBuilder`.

Давайте рассмотрим данный подход более подробно на примере интеграции базы данных GeneNet в среду BioUML.

Модуль GeneNet

База данных GeneNet содержит формализованное иерархическое описание структуры генных сетей и их компонентов (Kolpakov et al., 1998). База данных состоит из 12 таблиц, каждая из которых представлена в виде текстового файла. Таблицы *Cell*, *Compartment*, *Gene*, *Process*, *Protein*, *RNA*, и *Substance* содержат описание структурных компонентов генной сети, которые могут быть вершинами графов. Взаимодействия между компонентами описаны в таблице Relation. Структура диаграмм в специальном формате представлена в файле Schemes. Три вспомогательные таблицы Literature, Organism и Expert содержат список использованной литературы и видов организмов, а так же информацию об аннотаторах, осуществляющих ввод информации в базу данных .

Чтобы интегрировать базу данных GeneNet в среду BioUML, мы создали набор Java классов (package type на рис. 2) для представления информации из каждой таблицы GeneNet в объектно-ориентированном виде. Как правило, каждое поле база данных представляется в виде соответствующего атрибута Java класса. Другой набор классов (package access на рис. 2) обеспечивает трансформацию содержания базы данных в объекты соответствующих Java классов. Один вход из базы данных преобразуется в один Java объект.

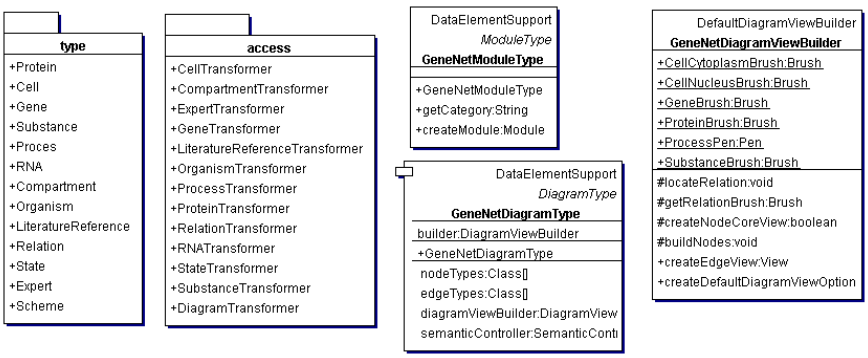

Рисунок 2. Структура (диаграмма классов) модуля GeneNet.

Чтобы обеспечить тот же самый способ графического представления, как в оригинальной версии компьютерной системы GeneNet (Kolpakov, Ananko, 1999), используется класс GeneNetDiagramViewBuilder.

Наконец, используя все эти классы, мы можем определить классы GeneNetDiagramType и GeneNetModuleType. После этого модуль GeneNet может быть интегрирован в среду BioUML и использоваться ее различными частями, например, программой просмотра диаграмм (рис. 3). На сайте http://www.biosoft.ru/biouml.shtml можно найти исходные

тексты для всех Java классов модуля GeneNet. Эти тексты могут использоваться разработчиками других баз данных по МБС для их интеграции в среду BioUML.

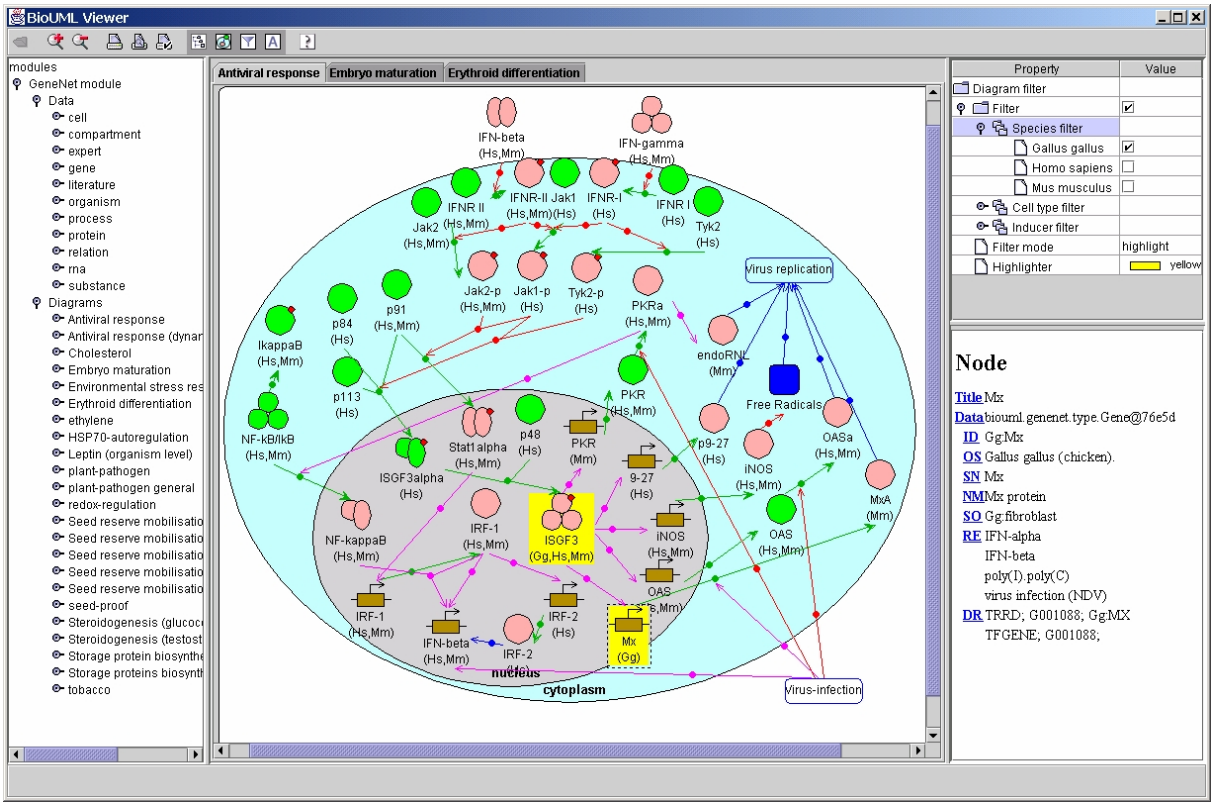

Рисунок 3. Внешний вид программы просмотра диаграмм BioUML, использующая модуль GeneNet. Слева – список таблиц и диаграмм модуля GeneNet; в центре – диаграмма “Antiviral response” из базы данных GeneNet; справа сверху – свойства фильтра для диаграммы (подсвечиваются вершины графа, удовлетворяющие условиям фильтра); справа снизу – описание одной из вершин графа (“Mx”).

Моделирование динамики МБС

Давайте рассмотрим как пример для моделирования динамики МБС простую двухкамерную фармокинетическую модель (Варфаломеев, Гуревич, 1999), где в первую камеру (кровь) одномоментно были введены 100 единиц некоторого лекарственного вещества A. Из крови вещество A лекарство может переноситься во вторую камеру (печень), где происходит его расщепление некоторым ферментом E с образованием продукта метаболизма B. Предположим, что скорость переноса лекарственного вещества A из крови в печень пропорциональна его количеству в крови с константой k_1, а скорость переноса из печени в кровь пропорциональна количеству A в печени с константой k_2. Так же предположим, что концентрация фермента E в печени неизменна и равна E_0, а динамика соответствующей реакции описывается уравнением Михаэлиса-Ментен с константой Km. Тогда динамика лекарственного количества вещества в крови (A_{blood}) и в печени (A_{liver}), а также динамика продукта метаболизма

(B_{liver}) может быть описана математической моделью, приведенной в таблице 1.

Таблица 1.

Математическая модель для описания динамики двухкамерной фармокинетической модели.

dy/dt	*Y(0)*
$\frac{dA_{blood}}{dt} = -k_1 A_{blood} + k_2 A_{liver}$ $\frac{dA_{liver}}{dt} = k_1 A_{blood} - k_2 A_{liver} - k_3 \frac{E_0 A_{liver}}{Km + A_{liver}/V_{liver}}$ $\frac{dB_{liver}}{dt} = k_3 \frac{E_0 A_{liver}}{Km + A_{liver}/V_{liver}}$	$A_{blood} = 100$ $A_{liver} = 0$ $B_{livver} = 0$

Используя данную математическую модель, исследователь может попытаться решить ее аналитически (если это возможно), либо написать программу для ее численного решения. Пример такой программы на языке MATLAB приведен на рис. 6.

Однако такой способ моделирования динамики МБС – построение математической модели вручную и последующая реализация этой модели в виде программы подходит только для сравнительно небольших и простых систем, как в описанном выше примере. Для более сложных систем такой подход, в лучшем случае, требует большого количества времени и внимания к деталям при выводе уравнений и написании соответствующей программы.

Визуальное моделирование

Задача моделирования динамики сложных МБС может быть существенно упрощена для исследователя при помощи компьютерных систем визуального моделирования. Графическое изображение систем в виде диаграмм представляет альтернативный синтаксис, позволяющий формально и полностью описать модель (Lee, 2001). Такой подход получил наибольшее распространение при моделировании физических и электротехнических систем. Разработано множество программных пакетов для визуального моделирования таких систем (Бенькович и др. 2002; Lee, 2001; и другие).

Для демонстрационных целей мы построили диаграмму (рис. 4) для описанной выше фармокинетической модели при помощи программы Simulink, входящей в состав пакета MATLAB. Возможно Simulink упрощает построение и анализ моделей МБС для опытных пользователей, однако используемый им визуальный синтаксис, как видно из

рисунка 4, плохо подходит для описания структуры МБС. Поэтому мы предлагаем новый визуальный синтаксис, ориентированный в первую очередь, на моделирование динамики МБС.

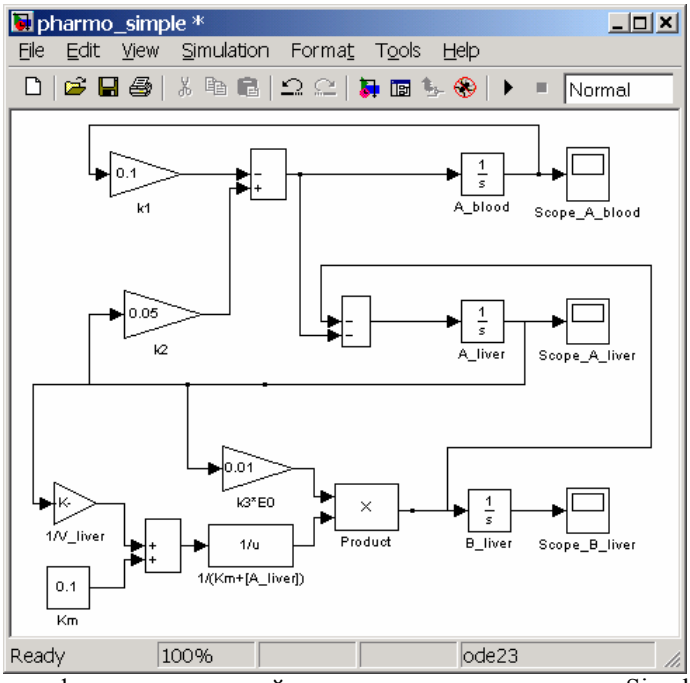

Fig. 4. Диаграмма фармокинетической модели, построенная в среде Simulink/MATLAB

BioUML modeler

Предлагаемый нами визуальный синтаксис для моделирования динамики МБС является расширением синтаксиса структурных диаграмм BioUML (см. пример структурной диаграммы на рис. 3). На этих диаграммах вершины графов выступают в качестве переменных, а правые части дифференциальных уравнений, описывающие изменение этих переменных, ассоциируются с ребрами графа. Пример такой диаграммы для описанной выше фармокинетической модели представлен на рисунке 5. BioUML modeler, являющейся частью среды BioUML, обеспечивает пользователя графическим интерфейсом для создания и редактирования графических моделей (рис. 5).

На основе графической модели BioUML modeler позволяет построить математическую модель МБС, представляемую в виде системы обыкновенных дифференциальных уравнений. На ее основе осуществляется генерация программы на языке MATLAB (рис. 6), который включает большой набор методов для численного решения обыкновенных дифференциальных уравнений (Shampine, Reichelt, 1997). BioUML modeler так же генерирует скрипт для системы MATLAB (рис. 6), позволяющий

запустить построенную модель и представить полученные результаты в графическом виде (рис. 7).

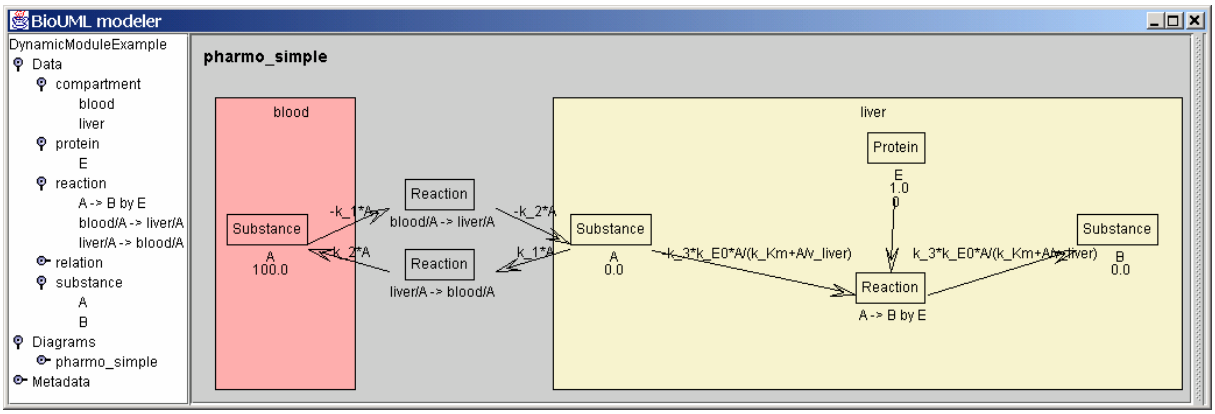

Рисунок 5. Графический интерфейс BioUML modeler. Строки над стрелками – правые части дифференциальных уравнений, связанные с ребрами графа; числа – начальные значения переменных, ассоциированных с вершинами графа.

```
%script for 'pharmo_simple' model simulation
%constants declaration
global k_1 k_2 k_3 k_E0 k_Km v_blood v_liver
k_1 = 0.1
k_2 = 0.05
k_3 = 0.01
k_E0 = 1
k_Km = 0.1
v_blood = 100
v_liver = 100

%Model variables and their initial values
 y = []
 y(1) = 100    % y(1) - blood/A
 y(2) = 0      % y(2) - liver/A
 y(3) = 0      % y(3) - liver/B

%numeric equation solving
[t,y] = ode23('pharmo_simple_dy',[0 200],y)

%plot the solver output
plot(t, y(:,1),'-', t,y(:,2),'--', t,y(:,3),'-.')
title ('Solving pharmo simple problem')
ylabel ('y(t)')
xlabel ('x(t)')
legend('blood/A','liver/A', 'liver/B');

--------------------------------------------------------------------------------
function dy = pharmo_simple_dy(t, y)
% Calculates dy/dt for 'pharmo_simple' model.

% constants declarations
global k_1 k_2 k_3 k_E0 k_Km v_blood v_liver

% calculates dy/dt for 'pharmo_simple' model
dy = [ -k_1*y(1) + k_2*y(2)
        k_1*y(1) - k_2*y(2) - k_3*k_E0*y(2)/(k_Km + y(2)/v_liver)
        k_3*k_E0*y(2)/(k_Km + y(2)/v_liver)]
```

Рисунок 6. Сгенерированные BioUML modeler M-файлы для численного моделирования фармокинетической модели 'pharmo_simple'. Сверху – скрипт для запуска модели и графического представления результатов рассчета; снизу – система дифференциальных уравнений (функция расчета dy/dt).

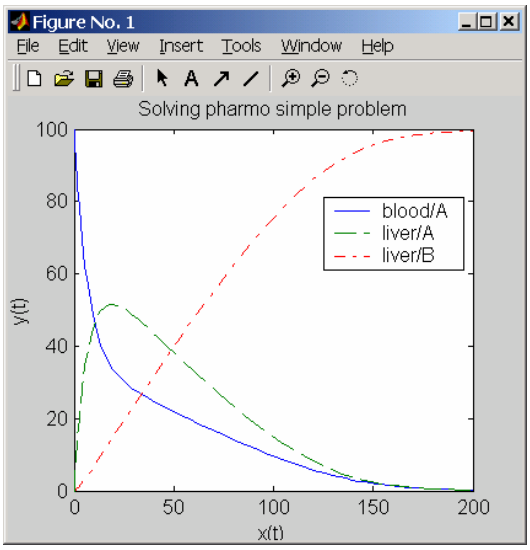

Рисунок 7. Результаты численного моделирования фармокинетической модели, полученные в результате запуска скрипта на рис. 6.

Заключение

На данный момент проект BioUML находится в стадии активной разработки. В данной работе мы описали основные идеи и продемонстрировали их работоспособность на примере работающего прототипа. Первая версия BioUML планируется на 3-ий квартал 2002 года. Среда BioUML будет свободно доступно для некоммерческих организаций. Мы так же планируем опубликовать основные исходные тексты, чтобы облегчить другим разработчикам интеграцию их модулей в среду BioUML.

На сайте http://www.biosoft.ru/biouml.shtml можно будет найти дополнительную информацию. Для обсуждения путей развития и использования BioUML научным сообществом существует специальный форум: http://groups.yahoo.com/group/biouml/.

Работа проводилась при частичной поддержке гранта Volkswagen-Stiftung (I/75941). Автор благодарен компании DevelopmentOnTheEdge.com за возможность использование их продукта BeanExplorer (www.beanexplorer.com) для создания графического интерфейса BioUML.

Список литературы

1. Бенькович Е.С., Колесов Ю.Б., Сениченков Ю.Б. (2002). *Практическое моделирование динамических систем*. БХВ-Петербург, СПб.
2. Варфаломеев С.Д., Гуревич К.Г. (1999) *Биокинетика*. FAIR-PRESS. М.

3. Kolpakov F.A., Ananko E.A., Kolesov G.B. and Kolchanov N.A. (1998) *GeneNet: a database for gene networks and its automated visualization*. Bioinformatics, **14(6)**, 529-537.
4. Kolpakov F.A., Ananko E.A. (1999) *Interactive data input into the GeneNet database*. Bioinformatics. **15(7-8)**:713-714 .
5. Lee E.A. (2001) *Overview of the Ptolemy Project*. Technical Memorandum UCB/ERL M01/11, University of California, Berkeley. (http://ptolemy.eecs.berkeley.edu/publications/papers/01/overview/)
6. OMG Unified Modeling Language Specification. http://www.omg.org/technology/documents/formal/uml.htm
7. Regiv A. (2002). *BioPathways Bibliography*. http://www.biopathways.org/2002/formalism_final.txt.zip
8. Schacherer F., Choi C., Gotze U., Krull M., Pistor S., Wingender E. (2001) *The TRANSPATH signal transduction database: a knowledge base on signal transduction networks*. Bioinformatics, **17(11)**, 1053-1057
9. Shampine L.F., Reichelt M.W. (1997). *The MATLAB ODE suite*. SIAM Journal of scientific Computing, **18(1)**, 1-22.