

# ИСПОЛЬЗОВАНИЕ SIMD-АРХИТЕКТУР В ПОСТРОЕНИИ ОБЪЕКТНО-ОРИЕНТИРОВАННЫХ МОДЕЛЕЙ ДИНАМИЧЕСКИХ СИСТЕМ

Баженов Л.А.

Кафедра ЭВМ, ДонГТУ  
e-mail:lechat@cs.dgtu.donetsk.ua

## Abstract

*Bazhenov L. Object Oriented approaches in modelling dynamic systems on SIMD computers. The Object Oriented approach is very fruitful in modelling and simulation. It's an important problem to design simulation tools based on object oriented concepts.*

## 1. Введение

Моделирование динамических систем с сосредоточенными (ДССП) и распределенными параметрами (ДСРП) является актуальной проблемой для всех областей техники и технологий. В последние годы методы и средства моделирования интенсивно используются в реальных технических и технологических проектах для проверки правильности проектных решений и выступают тем самым как важнейший фактор, гарантирующий качество проектирования, сокращение сроков внедрения проектов и освоения управляемых динамических систем обслуживающим персоналом. Играя столь важную роль в техническом и технологическом прогрессе, моделирование, как метод исследования, выдвигает ряд требований к вычислительным системам и их программному обеспечению как средствам реализации моделей. Анализ показал, что современным требованиям к средствам моделирования в наибольшей степени удовлетворяют массивно параллельные моделирующие среды (МПМС) [1,2,3,4,5], представляющие собой новую системную организацию аппаратно-программных средств, обеспечивающую моделирование ДССП, ДСРП реальной сложности. Теоретические и прикладные аспекты проблемы создания и применения МПМС включают компоновку эффективных архитектур параллельных вычислительных ресурсов, развитие системного и полнофункциональную разработку прикладного программного обеспечения моделирования ДССП и ДСРП.

Сложность как систем моделирования, так и самих моделей динамических систем (ДС) приводит к целесообразности использования объектно-ориентированного (ОО) подхода при разработке систем конструирования моделей с одной стороны [6,7,8] и проведения модельных экспериментов – с другой [9,10,11,12]. В данной статье автором рассмотрены аспекты использования объектно-ориентированной технологии на двух основных этапах: на этапе построения модели и на этапе проведения модельных экспериментов.

## 2. Концепции объектно-ориентированного проектирования в построении моделей динамических систем

Сложность разработки современных программных комплексов стимулирует поиск новых средств проектирования алгоритмов и программ. Бурно развивающийся в последнее время объектно-ориентированный подход [13,14,15] позволяет создавать системы высокой сложности. Построение моделей реальных ДС традиционно является сложной задачей. Таким образом, назрела проблема включения элементов объектно-ориентированного проектирования в системы моделирования. Рассмотрим основные концепции объектно-ориентированного подхода, которые можно считать первыми кандидатами на внедрение в моделирующие среды.

### 2.1. Абстрагирование

Для проектирования сложных систем используется метод декомпозиции, т.е. разработка системы переводится в плоскость разработки ее подсистем, которые, в свою очередь также могут быть декомпозированы. При ОО проектировании в системе выделяется множество объектов и строится диаграмма объектов [13]. Количество объектов и определение их взаимосвязей лежит на плечах проектировщика. Мощность множества выделяемых объектов зависит от уровня детализации, который регламентируется методом абстрагирования, определяющего глубину объектно-ориентированной декомпозиции.

Введение ОО абстракций в системы моделирования позволит не только автоматизировать процесс создания пользовательских предметных областей [7], но и определять сложные связи между объектами как одной, так и нескольких предметных областей. Таким образом, эксперт предметной области будет иметь возможность задавать также типы соединений, ограничивающие множество связывающихся объектов.

<b>Properties Section</b>
field#1 :: Property Description
field#2 :: Property Description
...
field#n :: Property Description
<b>Behavior Section</b>
method#1 :: Action Description
method#2 :: Action Description
...
method#k :: Action Description

Рисунок 1 – Спецификация интерфейса объекта

### 2.2. Инкапсуляция

Метод *инкапсуляции* вытекает из метода абстрагирования и состоит из сокрытия внутренней структуры объекта. Все объекты имеют возможность передавать сообщения друг другу, руководствуясь информацией об их интерфейсах. Интерфейс каждого объекта специфицируется описанием его свойств и поведения (рис.1). В объектно-ориентированном проектировании спецификация свойств и поведения однотипных объектов называется *классом*.

Секция описания свойств (property section) состоит из описания полей. Каждое поле хранит значение статической характеристики объекта. Поведение определяется набором методов (behavior section), каждый из которых задает одну из динамических характеристик объектов, характеристики, которая может изменять значение свойств.

Автоматизация процесса создания интерфейсов для объектов модели, т.е. генерация классов, позволит определять сложные, не стандартные поведенческие характе-

ристики объектов на стадии моделирования. В частности, методы могут быть заданы математическими выражениями, изменяющими содержимое полей, или же таблицами.

### 2.3. Наследование

*Наследование* – основа для создания новых предметных областей и расширение существующих. Принципы наследования, внедренные в системы моделирования, автоматизируют процесс заполнения фрейма рис.1 не с нуля, а с начальным множеством полей и методов, наследованных от класса-родителя.

### 2.4. Полиморфизм

*Полиморфизм* базируется на наследовании и позволяет производить однотипные операции над объектами различных классов. Основным условием применимости этого аспекта ОО проектирования является существование прямого или косвенного класса-родителя для классов, к которым принадлежат обрабатываемые объекты. Использование метода полиморфизма в системах моделирования позволит задавать специфическую обработку группы объектов, являющихся экземплярами различных классов.

### 2.5. Параллелизм

Аспект параллелизма относится к этапу моделирования. Само существование объектов вводит косвенный параллелизм. Тем не менее, следует предоставить эксперту предметной области автоматизированные средства формирования диаграммы процессов, если в этом есть необходимость.

Качество проведения модельных экспериментов находится в прямой зависимости от степени параллелизма, присутствующего как в программном коде, реализующем вычислительную модель, так и архитектуры вычислительной системы.

На рис.2 изображена диаграмма объектов. Для передачи информации между объектами используются сообщения. Если рис.2 представляет собой диаграмму объектов модели, то проведение модельных экспериментов может быть осуществлено с использованием МПМД- или SIMD-параллелизма [16,17,18].

Основой МПМД-архитектуры вычислительной системы является множество независимых процессоров, которые имеют возможность передавать данные друг другу посредством сообщений. Таким образом, применение объектно-ориентированного подхода к моделированию является естественным при реализации моделей на МПМД-архитектурах. При этом, каждый объект модели назначается отдельному процессору. Для повышения эффективности вычислений и минимизации количества передаваемой информации, одному процессору целесообразно назначать подмножество модельных объектов. При распределении объектов модели необходимо решать задачу балансировки нагрузки либо динамически (во время моделирования), либо статически (до запуска модели).

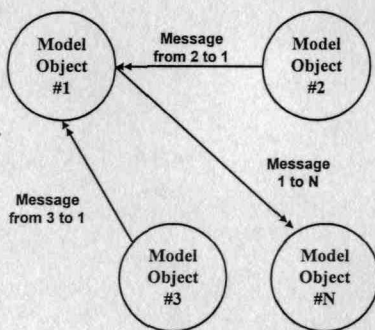


Рисунок 2 – Диаграмма объектов

Использование вычислительных систем SIMD-архитектуры привлекательно тем фактом, что SIMD-ЭВМ предоставляет большое количество (до 64К) дешевых процессорных элементов (ПЭ) и адаптирована для решения задач алгебры матриц, к которым сводятся большинство численных методов. Использование ОО подхода на этапе моделирования не типично для SIMD-архитектур, т.к. все множество ПЭ выполняет одну и ту же инструкцию, предоставленную host-процессором. Однако, модель разбита на непересекающиеся подмножества объектов, принадлежащих различным классам. Следовательно, имеется возможность производить параллельное вычисление выходных характеристик объектов по множествам. Рассмотрим использование SIMD-параллелизма при моделировании динамических систем с сосредоточенными параметрами.

### 3. ОО подход и SIMD-параллелизм

В [7,8] описываются дистрибутивные методы проведения модельных экспериментов. Их суть состоит в том, что каждый объект модели назначается отдельному процессорному элементу. Если SIMD-архитектура задается парой  $\Lambda = \langle \Omega, \mathfrak{Z} \rangle$ , где  $\Omega$  – множество процессорных элементов (ПЭ), топология связей которых задается отображением  $\mathfrak{Z}: \Omega \rightarrow \Omega$ , причем, отображение  $\mathfrak{Z}$  представлено решеткой, то процесс моделирования описывается следующими двумя шагами:

1. Просчет выходной информации объектов по классам, параллельно для каждой из групп объектов, принадлежащих одному классу.
2. Передача информации между объектами модели.

Будем использовать язык описания моделей, используемый в САУ и каждому блоку САУ сопоставим отдельный объект. Это низкоуровневое выделение объектов, основанное на операциях. Таким образом, получаем, в минимальном варианте, классы объектов, которые интегрируют, производят арифметические операции, вводят информацию и сохраняют результаты моделирования.

Проведем сравнительный анализ численных методов просчета систем обыкновенных дифференциальных уравнений (СОДУ), которые описывают модель САУ и дистрибутивного метода по количеству арифметических операций и операций передачи информации между объектами модели.

Рассмотрим метод Рунге-Кутты пятого порядка точности. Расчетные формулы СОДУ для каждого j-ого уравнения на i-ом шаге имеют следующий вид:

$$\begin{cases}
 k_{1j}^{(i)} = a_{j2}^{(i)} y_1^{(i)} \Delta t + a_{j2}^{(i)} y_2^{(i)} \Delta t + \dots + a_{jn}^{(i)} y_n^{(i)} \Delta t + f_j^{(i)} \Delta t; \\
 k_{2j}^{(i)} = a_{j2}^{(i+\frac{1}{2})} \left( y_1^{(i)} + \frac{k_{1j}^{(i)}}{2} \right) \Delta t + a_{j2}^{(i+\frac{1}{2})} \left( y_2^{(i)} + \frac{k_{1j}^{(i)}}{2} \right) \Delta t + \dots + a_{jn}^{(i+\frac{1}{2})} \left( y_n^{(i)} + \frac{k_{1j}^{(i)}}{2} \right) \Delta t + f_j^{(i+\frac{1}{2})} \Delta t; \\
 k_{3j}^{(i)} = a_{j2}^{(i+\frac{2}{2})} \left( y_1^{(i)} + \frac{k_{2j}^{(i)}}{2} \right) \Delta t + a_{j2}^{(i+\frac{2}{2})} \left( y_2^{(i)} + \frac{k_{2j}^{(i)}}{2} \right) \Delta t + \dots + a_{jn}^{(i+\frac{2}{2})} \left( y_n^{(i)} + \frac{k_{2j}^{(i)}}{2} \right) \Delta t + f_j^{(i+\frac{1}{2})} \Delta t; \\
 k_{4j}^{(i)} = a_{j2}^{(i+\frac{3}{2})} \left( y_1^{(i)} + k_{3j}^{(i)} \right) \Delta t + a_{j2}^{(i+\frac{3}{2})} \left( y_2^{(i)} + k_{3j}^{(i)} \right) \Delta t + \dots + a_{jn}^{(i+\frac{3}{2})} \left( y_n^{(i)} + k_{3j}^{(i)} \right) \Delta t + f_j^{(i+\frac{3}{2})} \Delta t; \\
 y_j^{(i+1)} = y_j^{(i+1)} + \frac{1}{6} \cdot (k_{1j}^{(i)} + 2k_{2j}^{(i)} + 2k_{3j}^{(i)} + k_{4j}^{(i)}).
 \end{cases} \tag{1}$$

Будем полагать, что каждое слагаемое правой части вычисляется на отдельном процессорном элементе, тогда количество операций сложения составит:

- 1.n операций при подсчете  $k_{1j}^{(i)}$ ;
- 2.3(n+1) операция при подсчете  $k_{2j}^{(i)}$ ,  $k_{3j}^{(i)}$ ,  $k_{4j}^{(i)}$ ;
- 3.4 операции при расчете  $y_j^{(i+1)}$ .

Итого имеем  $4n+7$  операций сложения.

Количество операций умножения, в силу последовательного вычисления коэффициентов, составляет  $2 \cdot 4 + 1 = 9$  операций.

Для дистрибутивных методов количество арифметических операций зависит от выбора метода интегрирования, т.к. интегрирование производится отдельным объектом. Для метода Симпсона, например, имеем:

$$\int_{t_i}^{t_{i+2}} x dt = \frac{h}{3} (x^{(i)} + x^{(i+1)} + x^{(i+2)}) \quad (2)$$

Для интегрирующего объекта, имеющего  $I$  входов, получаем следующую формулу:

$$\int_{t_i}^{t_{i+2}} x dt = \frac{h}{3} (x_1^{(i)} + x_2^{(i)} + \dots + x_1^{(i+1)} + x_2^{(i+1)} + \dots + x_1^{(i+2)} + x_2^{(i+2)} + \dots + x_1^{(i+2)}) \quad (3)$$

Таким образом, получаем  $I+1$  операций сложения и одну операцию умножения. Повышение точности дистрибутивного метода не повышает количество операций передачи информации, т.к. это изменение затрагивает только объекты интегрирования.

В табл.1 приведены вычисленные характеристики, полученные в результате сравнительного анализа численных и дистрибутивных методов.

Таблица 1

	Сложение	Умножение	Точность
Метод Эйлера	$n+1$	2	$O(h)$
Модифицированный Метод Эйлера	$2(n+1)$	4	$O(h^2)$
Метод Рунге-Кутты	$4n+7$	9	$O(h^5)$
Дистрибутивный метод третьего Порядка	4	1	$O(h^3)$
Дистрибутивный метод пятого Порядка	5	1	$O(h^5)$

Перейдем к сравнению количества операций передачи информации. В методе Рунге-Кутты для подсчета количества операций передачи информации заметим, что вычисление коэффициентов производится в четыре этапа, причем, необходимо производить передачу в процессе счета и в процессе распределения полученных значений, т.е. имеем  $4 \cdot (n^2 + n + 1)$ . Примем эту формулу как окончательную, пренебрегая передачами, необходимыми при подсчете  $u_j^{(i+1)}$ .

При подсчете количества операций передачи информации для дистрибутивного метода большее значение имеет способ назначения объектов процессорным элементам. Будем распределять объекты по сетке ПЭ следующим образом. Распределим цепочку связанных объектов, например, горизонтально. Далее, проходя по цепочке для каждого объекта, принадлежащего также и к другой цепочке, распределим последнюю перпендикулярно созданной первоначально цепочке. По ее завершению, необходимо совершить аналогичную процедуру для созданных перпендикулярных цепочек. Таким образом, будут созданы горизонтальные и вертикальные цепочки. При этом, достаточно совершить один сдвиг, чтобы передать вдоль цепочки выходную информацию на входы соответствующих объектов. Более того, этот сдвиг можно совершать одновременно для всех параллельных цепочек. На рис.3 приведены зависимости количества операций передачи информации от порядка СОДУ, полученные экспериментальным путем. Из гра-

фиков видно, что при повышении порядка системы выигрыш метода коммутационных плоскостей возрастает. Это достигается увеличением количества параллельных цепочек и их длины, по которым информацию можно передавать параллельно, осуществляя при этом один сдвиг. Следует также отметить, что повышение точности дистрибутивного метода не влияет на количество операций передачи информации, т.к. формула расчета усложняется лишь в объектах интегрирования.

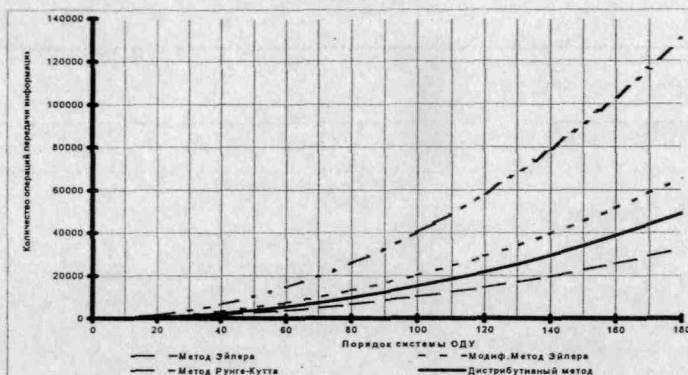


Рисунок 3 – Сравнительный анализ численных и дистрибутивного метода

## Заключення

Применение объектно-ориентированного подхода на этапе построения модели дает возможность гибко настраивать модельные предметные области, автоматизировать процесс описания группы объектов в виде классов, задавать параметры объектов. Реализация подобных систем должна базироваться на всем положительном опыте разработки фреймворков систем.

Использование дистрибутивного метода для проведения модельных экспериментов позволяет эффективно использовать ресурсы SIMD-ЭВМ, невзирая на то, что ОО подход естественен для MIMD-архитектур. Выигрыш предлагаемого дистрибутивного метода заключается в смещении акцентов с распределения уравнений СОДУ на распределение отдельных объектов по сетке ПЭ. Хотя экспериментальные результаты получены при выделении низкоуровневых модельных объектов, дистрибутивный метод применим и для более сложных объектов. При этом, также возможно распределить один объект на подмножество ПЭ при его относительно высокой вычислительной сложности. Интересной стороной дистрибутивного метода является тот факт, что задача повышения скорости вычислений сводится к топологическому исследованию, т.к. эффективность расчетов зависит от способа распределения объектов на сетке процессорных элементов.

## Література

- 1 Anoprijenko A., Bräunl T., Reuter A., Svjatnyj V., Zeitz M. *Massiv parallele Simulationsumgebung für dynamische Systeme mit konzentrierten und verteilten Parametern*. In G.Kampe, M.Zeitz (Hrsg): Tagungsband 9. Symposium Simulationstechnik ASIM'94 in Stuttgart, Verlag Vieweg 1994.
- 2 Anoprienko A., Braunl T., Feldmann L., Reuter A., Svjatnyj V., Zeitz M. *Massiv parallel models of net dynamic objects*. EUROSIM'95, Vienna, Elsevier 1995, pp. 237-242.
- 3 Svjatnyj V., Rasinkov V., Braeunl T., Reuter A., Zeitz M. *Problemorientierte massiv parallele Simulationsumgebung fuer dynamische Netzobjekte*. ASIM'96. Simulationstechnik. 10 Symposium in Dresden. Vieweg 1996.
- 4 Feldmann L., Svjatnyj V., Braeunl T., Reuter A., Zeitz M. *Implementierung und Effizienzanalyse von parallelen Simulationsverfahren fuer dynamische Systeme mit verteilten Parametern*. ASIM'97. Simulationstechnik. 11 Symposium in Dortmund. Vieweg 1997. s. 38-48.
- 5 Svjatnyi V., Feldman L., Lapko V., Anoprienko A., Reuter A., Braeunl T., Zeitz M. *Massive Parallel Simulation of Dynamic Systems*. Systemy Komputerowe i Sieci. Materialy Miedzynarodowej Konferencji Naukowej. Rzeszow, 26-27 czerwca 1997. p. 207-216.
- 6 Святный В.А., Баженов Л.А., *Принципы построения подсистемы диалога массивно параллельной моделирующей среды*. Сборник трудов факультета вычислительной техники и информатики. Выпуск 1. Донецк 1996. с. 24-31.
- 7 Anoprienko A., Bazhenov L., Bräunl T. *The Development of the Interface Subsystem for the Massive Parallel Simulation Environment*. ASIM'97. Simulationstechnik. 11 Symposium in Dortmund. Vieweg 1997.
- 8 Баженов Л.А. *Вопросы разработки SIMD-языка параллельного моделирования динамических систем*. Наукові праці Донецького Державного Університету. Серія: «Інформатика, кібернетика та обчислювальна техніка». Випуск 6. Донецьк 1996. с. 97-103.
- 9 Schmidt B. *Simulationssysteme der 5 Generation*. Simulation in Passau (SIP), Heft, 1, 1994, s. 5-6.
- 10 Schmidt B. *Die objektorientierte Modellspezifikation*. Simulation in Passau (SIP), Heft, 1, 1994, s. 7-8.
- 11 Schmidt B. *Das problem der Gleichzeitigkeit bei der objektorientierten Modellspezifikation*. Simulation in Passau (SIP), Heft, 2, 1995, s. 14-16.
- 12 Lang M. *Objektorientierte Simulationstechnik*. Universität Stuttgart, 1994, 29S.
- 13 Гради Буч. *Объектно-ориентированное проектирование*. Диалектика. 1992. с. 517.
- 14 Goad P., Yourdon E. *Object-Oriented Analysis*. Englewood Cliffs, NJ:Prentice-Hall. 1990.
- 15 Coad P., Yourdon E. *Object-oriented Design*. Englewood Cliffs. Prentice Hall, 1991.
- 16 Самофалов К.Г., Луцкий Г.М. *Структуры и организация функционирования ЭВМ и систем*. Киев. Выща школа, 1978.
- 17 *Супер ЭВМ: аппаратная и программная реализация*. Под редакцией Фернбаха С. Москва. Радио и связь, 1991.
- 18 *Параллельные вычисления*. Под редакцией Родрига Г. Москва. Наука, 1986.