

На правах рукописи

Бродский Юрий Игоревич

**ПРОБЛЕМА ОПИСАНИЯ И СИНТЕЗА
РАСПРЕДЕЛЕННЫХ ИМИТАЦИОННЫХ
МОДЕЛЕЙ СЛОЖНЫХ
МНОГОКОМПОНЕНТНЫХ СИСТЕМ**

Специальность: 05.13.17 Теоретические основы информатики

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
доктора физико-математических наук

Москва – 2016

Работа выполнена в Вычислительном центре им. А. А. Дородницына Федерального исследовательского центра «Информатика и управление» Российской академии наук (ВЦ ФИЦ ИУ РАН).

Научный консультант д.ф.-м.н., профессор, член-корр. РАН, Павловский Юрий Николаевич.

Официальные оппоненты: д.ф.-м.н., профессор, Афанасьев Александр Петрович, зав. Центром распределенных вычислений в Федеральном государственном бюджетном учреждении науки Институт проблем передачи информации им. А.А. Харкевича Российской академии наук (ИППИ РАН);

д.ф.-м.н., профессор, Михайлов Александр Петрович, зав. отделом в Федеральном государственном бюджетном учреждении науки Институт прикладной математики им. М.В. Келдыша Российской академии наук (ИПМ РАН);

д.ф.-м.н., профессор, член-корр. РАН, Петров Игорь Борисович, зав. кафедрой Информатики в Федеральном государственном образовательном учреждении высшего профессионального образования «Московский физико-технический институт (государственный университет)» (МФТИ).

Ведущая организация: Федеральное государственное учреждение «Федеральный научный центр Научно-исследовательский институт системных исследований Российской академии наук» (ФГУ ФНЦ НИИСИ РАН).

Защита состоится _____ 2016 г. в _____ час. _____ мин. на заседании диссертационного совета Д 002.073.05 при Федеральном государственном учреждении «Федеральный исследовательский центр Информатика и управление» Российской академии наук, по адресу: 119333, г. Москва, ул. Вавилова, д. 40.

С диссертацией можно ознакомиться в библиотеке и на сайте ФИЦ ИУ РАН – <http://web.frcsc.ru/diss-council/00207305/diss>

Автореферат разослан _____ 2016 г.

Ученый секретарь

диссертационного совета,

д.ф.-м.н., профессор

Рязанов В.В

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность исследования. Моделирование – один из основных способов познания мира. Имитационное моделирование сложных систем – часто единственный способ исследования их поведения и возможностей, например, в случаях, когда учиться на ошибках, допущенных при создании реальных систем, было бы непозволительно дорого. Работа посвящена проблемам имитационного моделирования достаточно широкого класса сложных систем, который характеризуется атомистическим и фрактальным устройством своих представителей. Такое понимание сложной системы отражено, например, в одноименной статье Н.П. Бусленко в БСЭ: «Сложная система – составной объект, части которого можно рассматривать как системы, закономерно объединённые в единое целое в соответствии с определенными принципами или связанные между собой заданными отношениями». В дальнейшем, под сложными системами будем понимать именно системы, состоящие в свою очередь также из сложных систем. При этом считается, что хорошо известно устройство «атомов» системы, их поведение, а также закономерности их взаимодействия между собой. Задачей имитационного моделирования в данном случае является воспроизведение поведения всей системы в целом, с целью выяснения ее возможностей, особенностей, возможно решению относительно нее каких-либо управлеченческих или оптимизационных задач.

Эта задача далеко не тривиальна. Достаточно сложной является даже входящая в нее задача описания всего, что известно о сложной системе. Тем более сложной является компьютерная реализации модели сложной системы.

Одним из магистральных направлений развития современной информатики являются распределенные и параллельные вычисления. Создание инструментальной системы распределенного имитационного моделирования в интернете позволило бы сообществу исследователей, занимающихся моделированием,

обмениваться результатами своей работы «не сходя с места», а также разрабатывать совместные модели.

Возникают следующие относящиеся к теоретическим основам информатики задачи:

- Разработка принципов создания языков описания данных, средств представления знаний, и методов взаимодействия информационных процессов. (пп. 3, 4, 12)
- Разработка теоретических основ создания программных систем для такой новой информационной технологии, как распределенное имитационное моделирование. (п. 14)
- Разработка новых интернет-технологий средств интеллектуализации бизнес-процессов (распределенное имитационное моделирование). (п. 9)

В настоящее время существует ряд инструментальных средств имитационного моделирования, в том числе, и распределенного. Их обзору и подробному анализу посвящена вторая глава диссертации. Однако большинство из этих средства основаны на эвристических способах, пусть даже и весьма успешных, решения проблем, возникающих при моделировании.

Цель работы. Формализовать понятие модели сложной системы (в указанном выше смысле). На основе этой формализации найти способ описания модели сложной системы, отражающий постулированное выше знание о ее составе, свойствах и умениях ее компонент, их связей между собой и способах взаимодействия. Теоретически исследовать вопрос возможности построения синтеза поведения модели сложной системы на основе такого ее описания. Построить систему автоматического синтеза поведения модели сложной системы по ее описанию, и на этой основе – инструментальную систему распределенного имитационного моделирования.

Метод исследования. Для формализации и анализа процесса моделирования сложной системы применялись методы математического моделирования, теория родов структур Н. Бурбаки, методы геометрической теории декомпозиции, методы

объектного анализа. Полученные результаты были применены для разработки концепции моделирования, которая, в свою очередь, была реализована в макете рабочей станции пииринговой сети моделирования в Интернете.

Научная новизна.

- Новой является предложенная концепция модельного синтеза и парадигма модельно-ориентированного программирования – технологии описания, синтеза и реализации имитационных моделей сложных многокомпонентных систем.
- Новой является формализация понятия модели сложной системы – семейство родов структур «модель-компонент» – математические объекты, похожие на объект объектного анализа тем, что имеют характеристики и методы, однако, отличающиеся от него тем, что имеют поведение, – т.е. способность стандартным заданным образом отвечать на стандартные запросы внутренней и внешней среды.
- Новым является выделение класса имитационных моделей, к которым применимы методы модельного синтеза – класс моделей с кусочно-гладкой траекторией с не более чем конечным числом разрывов первого рода и непрерывной слева в любой точке отрезка моделирования, кроме начала.
- Новой является предлагаемая архитектура пииринговой сети распределенного имитационного моделирования, где каждая рабочая станция, несмотря на то, что может быть абсолютно тонким клиентом, всегда остается центром моделирования и полным хозяином построенной модели (пусть даже состоящей исключительно из «чужих» составляющих).

Практическая и теоретическая ценность. Результаты работы применялись:

- При создании ряда моделей имитации вооруженной борьбы в космосе.
- В рамках программы фундаментальных исследований Президиума РАН «Разработка фундаментальных основ созда-

ния научной распределенной информационно-вычислительной среды на основе технологий GRID».

- При выполнении поддержанных РФФИ проектов 07-07-00071-а «Разработка распределенной инструментальной системы имитации», 10-07-00176-а «Разработка инструментальной системы для создания проблемно-ориентированных распределенных вычислительных комплексов», 10-07-00420-а «Исследование процессов глобализации с использованием распределенной эколого-демографо-экономической модели» и 13-01-00499-а «Методы геометрической теории декомпозиции в описании и синтезе сложных многокомпонентных имитационных моделей», а также поддержанного РГНФ проекта 12-06-00932-а «Имитационное моделирование динамических изменений социально-демографической структуры населения регионов с учетом экологической и экономической специфики».
- В ряде учебных пособий и учебных курсов по математическому, имитационному и компьютерному моделированию.

Достоверность результатов обусловлена строгостью примененного математического аппарата и подтверждена рядом практических реализаций в инструментальной системе распределенного имитационного моделирования и имитационных моделях.

На защиту выносятся следующие основные результаты и положения:

- Концепции модельного синтеза и модельно-ориентированного программирования.
- Формализация понятия модели сложной системы. Понятие модели-компоненты.
- Язык описания компонент и комплексов (ЯОКК).
- Архитектура пиринговой сети имитационного моделирования в Интернете и первый этап ее реализация в виде макета рабочей станции этой сети.

Апробация работы. Результаты работы докладывались и обсуждались на VII Всероссийской школе-семинаре "Прикладные проблемы управления макросистемами" (Апатиты, 31 марта - 4 апреля 2008 года.), на IV Всероссийской научной конференции «Математическое моделирование развивающейся экономики и экологии», ЭКОМОД-2009 (Киров, 6-12 июля 2009 г.), на Третьей международной конференции «Математическое моделирование социальной и экономической динамики» MMSED-2010 (Москва, 23-25 июня 2010г.), на VI Московской международной конференция по исследованию операций ORM2010: (Москва, 19-23 октября 2010г.), на Международной конференции по прикладной математике и информатике, посвященной 100-летию со дня рождения академика А. А. Дородницына (ВЦ РАН, Москва, 7-11 декабря 2010г.), на Международных суперкомпьютерных конференциях «Научный сервис в сети Интернет: поиск новых решений», 7 – 22 сентября 2012 г. и «Научный сервис в сети Интернет: все грани параллелизма», 23-28 сентября 2013, Абрау-Дюрсо, на Третьей международной конференции по высокопроизводительным вычислениям НРС-УА-2013, 7 – 11 октября 2013 г., Киев, на Шестой всероссийской научно-практической конференции «Имитационное моделирование. Теория и практика» ИММОД-2013, 16-18 октября 2013 г., Казань.

Результаты работы вошли в ряд учебных пособий и в содержание учебных курсов по математическому, имитационному и компьютерному моделированию, читавшихся соискателем в течение ряда лет в МПГУ, МГТУ им. Н.Э. Баумана, МГУ им. М.В. Ломоносова, и его коллегами в МГУ им. М.В. Ломоносова, МФТИ (ГУ) и РХТУ им. Д.И. Менделеева.

Публикации. Основные результаты работы опубликованы в 15 публикациях в изданиях, рекомендованных ВАК или приравненных к таковым, 9 монографиях, 4 учебных пособиях, 2 статьях в энциклопедии EOLSS и еще в ряде публикаций, общий список которых приводится в конце автореферата.

Личный вклад соискателя.

- Разработка концепции модельного синтеза и модельно-ориентированного программирования – технологии описания, синтеза и реализации имитационных моделей сложных многокомпонентных систем.
- Формализация основных понятий описания и синтеза моделей сложных систем.
- Разработка языка описания комплексов и компонент (ЯОКК).
- Разработка концепции инструментальной системы распределенного имитационного моделирования.
- Разработка архитектуры пиринговой сети имитационного моделирования.
- Реализация программного обеспечения макета рабочей станции сети имитационного моделирования.

Структура и объем работы. Работа состоит из введения, трех глав, заключения, приложения и списка литературы, изложена на 343 страницах рукописи. Библиография включает 90 названий.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во **введении** ставится проблема описания, синтеза и реализации имитационных моделей сложных многокомпонентных систем, обосновывается актуальность этой проблемы, дается краткая схема ее решения в рамках данной работы и обзор примененных для этого решения методов.

Первая глава представляет собой необходимый для построений третьей главы краткий, написанный с позиций обычной (небурбаковской) аксиоматики, реферат работ Н. Бурбаки об основах теории родов структур и работ Ю.Н. Павловского о началах геометрической теории декомпозиции.

Выбор теории родов структур Н. Бурбаки и теории геометрической декомпозиции Ю.Н. Павловского в качестве языковой среды данной работы обусловлен адекватностью и удобством

этой среды, во-первых, для формализации основных понятий и методов развивающегося далее модельного синтеза, и, во-вторых, для сравнения основных методов и средств модельного синтеза и модельно-ориентированного программирования с одной стороны, и объектного анализа и объектно-ориентированного программирования – с другой.

Род структуры объявляет о том, что математический объект будет состоять из частей $\sigma_1, \dots, \sigma_r$ некоторых множеств. Это записывается следующим образом:

$$\sigma_1 \subset S_1(X_1, \dots, X_n, A_1, \dots, A_m), \dots, \sigma_r \subset S_r(X_1, \dots, X_n, A_1, \dots, A_m).$$

Здесь множества X_1, \dots, X_n называются основными базисными, множества A_1, \dots, A_m – вспомогательными базисными.

Множества $S_k(X_1, \dots, X_n, A_1, \dots, A_m)$, $1 \leq k \leq r$, – так называемые ступени, построенные по схеме S_k из базисных и вспомогательных множеств $X_1, \dots, X_n, A_1, \dots, A_m$. Ступени получаются путем применения к исходным множествам и/или уже имеющимся ступеням операций декартова произведения \times и взятия множества всех подмножеств $\beta(\cdot)$. А именно:

1. По определению, X_i – ступень при любом $1 \leq i \leq n$. Аналогично, A_j – ступень при любом $1 \leq j \leq m$.
2. Если S – ступень, то и $\beta(S)$ – ступень.
3. Если S и S' – ступени, то и $S \times S'$ – ступень.
4. Других ступеней нет.

Схема S_k фиксирует исходные множества и порядок применения к ним двух указанных выше операций.

Основные и вспомогательные базисные множества играют разную роль в построениях родов структур (например, в построениях данной работы вспомогательные множества вообще не используются). Основные базисные множества должны быть обозначены разными буквами. На обозначения вспомога-

тельных множеств таких ограничений не накладывается. Соотношения

$$\sigma_1 \subset S_1(X_1, \dots, X_n, A_1, \dots, A_m), \dots, \sigma_r \subset S_r(X_1, \dots, X_n, A_1, \dots, A_m)$$

называются соотношениями типизации. В род структур, кроме соотношений типизации может входить еще некоторое соотношение

$$R(X_1, \dots, X_n, \sigma_1, \dots, \sigma_r, A_1, \dots, A_m, \xi_1, \dots, \xi_s).$$

Это соотношение предъявляет к роду структуры некоторые требования. Здесь ξ_1, \dots, ξ_s – соотношения, касающиеся множеств A_1, \dots, A_m . Вспомогательные множества A_1, \dots, A_m и соотношения ξ_1, \dots, ξ_s не преобразуются – отображения рассматриваются только над базисными множествами.

Соотношение $R(X_1, \dots, X_n, \sigma_1, \dots, \sigma_r, A_1, \dots, A_m, \xi_1, \dots, \xi_s)$ называется «аксиомой» данного рода структуры. К нему предъявляется требование: оно должно быть переносимо при биекциях. Это означает, что

$$(f_i; X_i \rightarrow X'_i, i = 1, 2, \dots, n,) - \text{биекции} \Rightarrow \\ (R(X, \sigma, A, \xi) \Rightarrow R(X', \sigma', A, \xi)), \text{ где } \sigma' = S(f, id_A).$$

Здесь $S(f, id_A)$ – так называемое «распространение» отображений f базисных множеств и тождественное распространение вспомогательных множеств по схеме S .

Род структуры будет обозначаться далее следующим образом:

$$\Sigma(A_1, \dots, A_n, \xi_1, \dots, \xi_s) = \\ = < X_1, \dots, X_n; [\sigma_1 \subset S_1(X, A), \dots, \sigma_r \subset S_r(X, A)]; [R(X, \sigma, A, \xi)] >$$

или более коротко:

$$\Sigma[(A, \xi)] = < X; [\sigma \subset S(X, A)]; [R(X, \sigma, A, \xi)] >.$$

Ломаные скобки «<» и «>» здесь играют роль ограничителя. Необязательные элементы рода структуры, как это принято при записи такого сорта конструкций, поставлены в квадратные скобки. Далее по возможности будет использоваться короткая форма всех соотношений.

Пусть имеются множества (E, τ) и выполняются соотношения $\tau \subset S(E, A)$ и $R(E, \tau, A, \xi)$, т. е. соотношение типизации и аксиома рода структуры $\Sigma[(A, \xi)]$ для множеств (E, τ) . Тогда говорят, что объект (E, τ) снабжен структурой τ рода $\Sigma[(A, \xi)]$ или что τ есть структура рода $\Sigma[(A, \xi)]$ на множестве E . Обозначение $\Sigma[(A, \xi)]$ содержит то, что не меняется при переходе от одного $\Sigma[(A, \xi)]$ -объекта к другому.

Это означает, что множества A_1, \dots, A_m и соотношения ξ_1, \dots, ξ_s являются одними и теми же для всех объектов данного рода структуры.

Тем самым, как уже говорилось, математические объекты делятся на классы. К классу относятся объекты, снабженные структурой данного рода $\Sigma(A, \xi)$.

Вторая глава посвящена обзору ряда существующих инструментальных систем имитационного моделирования. Рассматриваются: старейшая, но до сих пор достаточно популярная система GPSS (General Purpose Simulation System – система имитации общего назначения). Далее рассматривается разработанная в конце 80-х гг. в ВЦ АН СССР многоязыковая инструментальная система моделирования MISS (Multilingual Instrumental Simulation System). Затем – разработанная во второй половине 90-х по заказу министерства обороны США и ставшая в 2000г. международным стандартом, спецификация архитектуры высокого уровня распределенного моделирования HLA (High Level Architecture). Рассматривается Унифицированный язык моделирования (UML), предназначенный для описания, визуализации и документирования объектно-ориентированных систем и бизнес-процессов. Далее рассмат-

ривается инструментальная система моделирования AnyLogic – не только качественный инструмент моделирования, но и успешный коммерческий проект российской фирмы XJ Technologies, впервые представившей этот продукт в 2000г. Наконец, завершается обзор знакомством с рядом свободно распространяемых инструментальных средств мультиагентного моделирования, обычно относимым к системам ABMS (Agent-Based Modeling and Simulation).

Основные выводы обзора:

Анализ приведенных примеров реализаций инструментальных средств моделирования показывает, что авторы этих разработок в разное время, и по всей видимости независимо друг от друга, часто приходили к весьма сходным решениям относительно некоторых способов организации вычислительного процесса имитации. Например, большинство приведенных в обзоре систем четко ориентированы на парадигмы объектного анализа и объектно-ориентированного программирования. Однако крайне редко, практически никогда не приводилось обоснование, почему что-либо делается так, а не иначе. Возникает вопрос, действительно ли предлагаются наилучшие решения из всех возможных в данной области? Или же среди них есть случайные, без которых вполне можно было бы обойтись?

Также редки попытки четко описать классы задач, к которым применима та или иная инструментальная система имитации.

В следующей главе делается попытка ответить на эти вопросы.

Третья глава посвящена модельному синтезу и модельно-ориентированному программированию – предлагаемой в работе технологии описания, синтеза и реализации имитационных моделей сложных многокомпонентных систем.

Прежде всего, обсуждается гипотеза о замкнутости модели. Гипотеза о замкнутости в имитационном моделировании есть на самом деле выражение уверенности разработчика в способности создать адекватную имитационную модель: чтобы взяться за ее построение, нужно быть уверенным в успехе. Бу-

дем считать, что в любой момент времени мы умеем узнавать значения внешних характеристик модели, и предполагать, что знания значений внутренних и внешних характеристик модели в момент t достаточно для вычисления ее внутренних характеристик на некотором интервале $(t, t + \Delta t)$.

Для моделей сложных систем предположение о непрерывной зависимости характеристик от времени не является естественным. Очень часто здесь зависимости некоторых характеристик от времени явно носят дискретный характер. Кроме того, все вычисления характеристик модели осуществляются, как правило, с помощью компьютера. Попробуем обсудить возможности компьютерного построения моделей сложных много-компонентных систем.

Отметим, что реализуя имитационные вычисления на компьютере, за конечное время мы сможем обработать лишь конечное число разрывов первого рода. Поэтому будем искать траекторию нашей модели $\vec{X}(t)$ в классе кусочно-гладких по t функций. Далее, поскольку шаг моделирования Δt выбирается так, что на протяжении этого шага изменения модели не слишком заметны, будем всегда относить разрыв первого рода на начало шага, а далее на протяжении Δt будем считать траекторию модели $\vec{X}(t)$ гладкой. Траекторию модели будем вычислять с некоторой заданной точностью $\varepsilon > 0$. Например, если траектория модели определяется дифференциальным уравнением $\dot{X} = F(X, a)$, то при малых $\tau > 0$ справедливо

$$X(t + \tau) \approx X(t) + F(X(t), a(t))\tau.$$

Попытаемся теперь построить траекторию модели $\vec{X}(t)$ на некотором макроскопическом отрезке модельного времени $[0, T]$, попутно выясняя, в каких предположениях может быть успешным такое построение.

Определение 1

Будем называть модель замкнутой в точке $t \in [0, T]$, если найдется число $\Delta t > 0$, $t + \Delta t \in (0, T]$, которое будем называть отрезком прогноза модели для точки t , такое что:

1. На основании внутренних и внешних характеристик модели $\vec{X}(t)$ и $\vec{a}(t)$ можно определить, есть ли в точке t разрыв траектории $\Delta\vec{X}(t)$, и если он есть, – вычислить его.
2. Далее, на полуинтервале $(t, t + \Delta t]$, траектория модели, выходящая из точки $\vec{X}(t) + \Delta\vec{X}(t)$, является гладкой функцией времени, например, решением системы дифференциальных уравнений $\dot{\vec{X}} = F_t(\vec{X}, a)$.

Определение 2

Будем называть модель локально замкнутой на отрезке $[0, T]$, если она замкнута в любой точке $t \in [0, T]$.

Определение 3

Будем называть модель прогнозируемой или лапласовской на отрезке $[0, T]$, если существует конечное разбиение этого отрезка точками $0 = t_0 < t_1 < \dots < t_n = T$, такое что модель замкнута в каждой из точек t_{i-1} , $i = 1, \dots, n$, и каждый из полуинтервалов $(t_{i-1}, t_i]$, $i = 1, \dots, n$, принадлежит отрезку прогноза для своего левого конца.

Очевидно, если модель прогнозируема в смысле последнего определения и удалось найти соответствующее разбиение отрезка $[0, T]$ точками $0 = t_0 < t_1 < \dots < t_n = T$, то задачу построения модели можно считать в принципе решенной. Отметим также, что если модель прогнозируется на отрезке, переопределяя ее траекторию не более чем в конечном числе точек разрыва, можно считать, что эта траектория непрерывна слева в любой точке отрезка.

С другой стороны, если не выполнены условия определения 2, т. е. существуют точки $t \in [0, T]$, откуда невозможно сделать даже малый шаг на $\Delta t > 0$, то задача построения модели представляется безнадежной. Поэтому требование локальной замкнутости на отрезке будет одним из базовых требований, предъявляемых к модели.

Из локальной замкнутости на отрезке, прогнозируемость на этом отрезке, вообще говоря, не следует.

Предложение 1

Сделать локально замкнутую на отрезке $[0, T]$ модель прогнозируемой на этом отрезке может добавление требования непрерывности слева траектории модели $\vec{X}(t)$ в любой точке $t \in (0, T]$. ■

Из сказанного выше вытекает ряд весьма важных следствий, которые будут перечислены далее:

1. Коль скоро мы признаем на каждом отрезке прогноза $[t, t + \Delta t]$ функциональную зависимость от значений характеристик модели в момент t как величины возможного скачка траектории в момент t , так и последующей ее гладкой динамики на $(t, t + \Delta t]$, – естественно признать и возможность вычисления этих функциональных зависимостей в функциональной парадигме программирования.
2. Функциональная зависимость однозначна по определению. Отсюда следует, что если по каким-то соображениям (например, происходящим из предметной области моделирования, или из способа организации вычислительного процесса), нам представляется удобным декомпозировать эту зависимость на ряд параллельно выполняемых вычислений, то эта декомпозиция не должна нарушать упомянутой однозначности. Таким образом, если два параллельных процесса пытаются одновременно модифицировать одну и ту же характеристику модели – это есть просто ошибка ее проектировщика – нарушение однозначности функцио-

нальной зависимости (а отнюдь не отражение объективной сложности модели или моделируемой ею системы). Стало быть, декомпозированные на параллельные процессы вычисления функциональной зависимости, не эквивалентны исходным. Сохранение однозначности вычислений – необходимое условие такой эквивалентности.

Из определения 3 прогнозируемой или лапласовской модели следует, что процесс вычисления ее траектории естественным образом декомпозируется на чередующиеся вычисления «скаков» и интервалов гладкой динамики. Траекторию модели можно считать непрерывной слева. Разрывов траектории модели первого рода может быть лишь конечное число, они происходят мгновенно в модельном времени и зависят только от предела слева характеристик модели в момент разрыва. На интервалах гладкой динамики внутренние характеристики модели непрерывно зависят от ее характеристик на левом конце интервала и от времени прошедшего от начала интервала. Именно эта декомпозиция и определяет предлагаемые в ниже правила организации имитационных вычислений (аксиому поведения).

Далее дается формализованное описание элементарной модели сложной системы – модель-компоненты.

Введем однопараметрическое семейство родов структур Σ_N «модель-компонента». Параметром N семейства Σ_N является количество процессов модели-компонента. Формально процессы модели-компонента будут определены ниже соотношениями типизации (10) и аксиомами R_9 .

$$\Sigma_N = \langle X, M, E, \{M_j\}_{j=1}^N, \{E_j\}_{j=1}^N; \quad$$

$$x \subset X,$$

$$a \subset X,$$

$$s \subset M,$$

$$f \subset M,$$

$$\{m_{j,real} \subset M_j \times M\}_{j=1}^N,$$

$$\{e_{j,real} \subset E_j \times E\}_{j=1}^N,$$

$$\{m_{j,in} \subset M_j \times \beta(X)\}_{j=1}^N,$$

$$\{m_{j,out} \subset M_j \times \beta(X)\}_{j=1}^N,$$

$$\{e_{j,in} \subset E_j \times \beta(X)\}_{j=1}^N,$$

$$\{m_j^0 \subset M_j\}_{j=1}^N,$$

$$\{sw_j \subset E_j \times M_j \times M_j\}_{j=1}^N,$$

$$\{p_j \subset \beta(M_j) \times \beta(E_j) \times M_j \times \beta(E_j \times M_j \times M)\}_{j=1}^N;$$

$$R_1: (x \cup a = X) \& (x \cap a = \emptyset),$$

$$R_2: (s \cup f = M) \& (s \cap f = \emptyset),$$

$$R_3: \{(\forall m \in M_j)(\exists !\tilde{m} \in M)(\{m, \tilde{m}\} \in m_{j,real})\}_{j=1}^N,$$

$$R_4: \{(\forall e \in E_j)(\exists !\tilde{e} \in E)(\{e, \tilde{e}\} \in e_{j,real})\}_{j=1}^N,$$

$$R_5: \{(\forall m \in M_j)(\exists !r \in \beta(X))(\{m, r\} \in m_{j,in})\}_{j=1}^N,$$

$$R_6: \{(\forall m \in M_j)(\exists !r \in \beta(x))(\{m, r\} \in m_{j,out})\}_{j=1}^N,$$

$$R_7: \{(\forall e \in E_j)(\exists !r \in \beta(X))(\{e, r\} \in e_{j,in})\}_{j=1}^N,$$

$$R_8: \left\{ \left(\forall e \in E_j \right) \left(\exists ! r \in M_j \times M_j \right) \left(\{e, r\} \in sw_j \right) \right\} \& \\ \& \left(\left(\{e, r\} \in sw_j, \{\tilde{e}, \tilde{r}\} \in sw_j, r = \tilde{r} \right) \Rightarrow (e = \tilde{e}) \right) \}_{j=1}^N$$

$$R_9: \left\{ p_j = \left\{ M_j, E_j, m_j^0, sw_j \right\}_{j=1}^N \right\},$$

R_{10} : аксиома однозначности вычисления характеристик модели-компоненты,

R_{11} : аксиома поведения модели-компоненты (организации имитационных вычислений).

Обозначение $\left\{ \dots_j \right\}_{j=1}^N$ используется для краткости и означает, что содержимое скобок повторяется через запятую N раз, при этом индекс j заменяется на $1, \dots, N$. Например, $\left\{ M_j \right\}_{j=1}^N$ есть краткий вариант записи M_1, \dots, M_N .

Дадим пояснения приведенному выше формальному определению рода структуры «модель-компоненты». Оно заключено в угловые скобки «<» и «>» и состоит из трех частей, которые отделяются друг от друга символом «;», а отдельные определения в каждой части разделяются символом «,». В первой части определяются базисные множества. Во второй части перечислены соотношения типизации, которые определяют родовые константы, как части множеств, построенных из базисных специальным образом – возможным последовательным применением операций декартова произведения \times и взятия множества всех подмножеств $\beta(\cdot)$. В третьей части перечислены аксиомы рода структуры $R_1 - R_9$, которые устанавливают определенные зависимости между базисными множествами и родовыми константами.

В качестве основных базисных множеств выбраны

$$X, M, E, \left\{ M_j \right\}_{j=1}^N, \left\{ E_j \right\}_{j=1}^N$$

Здесь X – множество характеристик модели, причем иногда будем разделять его на два подмножества: $X = \{x, a\}$, где x – внутренние характеристики модели, то, что в соответствии с гипотезой о замкнутости полностью определяет ее состояние, а a – ее внешние характеристики, то, что в силу той же гипотезы о замкнутости полностью определяет взаимодействие модели с внешним по отношению к ней миром. Первые два из соотношений типизации

$$x \subset X, a \subset X, \quad (1)$$

утверждают, что внутренние характеристики модели x и внешние характеристики a – есть части ее характеристик. Аксиома R_1 : $(x \cup a = X) \& (x \cap a = \emptyset)$ утверждает, что эти части x и a не пересекаются, и в совокупности составляют все множество характеристик модели X .

Далее идет M – множество различных реализаций методов-элементов, элементарных умений нашей модели, среди которых мы также иногда будем выделять два подмножества: $M = \{s, f\}$ – медленных s , реализующих гладкую зависимость внутренних характеристик модели от ее внутренних и внешних характеристик, если например, $\dot{x} = F(x, a)$, то $x(t + \Delta t) \approx x(t) + F(x(t), a(t))\Delta t$, и быстрых f – реализующие скачки внутренних характеристик модели: $\Delta x = G(x(t), a(t))$. Точно так же, как и в случае с характеристиками, запишем соответствующие соотношения типизации:

$$s \subset M, f \subset M, \quad (2)$$

и аксиому: R_2 : $(s \cup f = M) \& (s \cap f = \emptyset)$.

Далее, E – множество различных реализаций методов-событий, связанных с моделью. События – это то, на что обязана реагировать наша модель. Метод-событие – функция, которая, получая на входе подмножество характеристик $Y \subset X$, на выходе дает неотрицательное число, означающее, что собы-

тие наступило, если число равно нулю, или же прогноз времени до наступления события, если число положительно.

Отметим, что во множествах X, M, E нет одинаковых элементов. Этот факт также утверждается аксиомой рода структуры R_{10} . Для X это следует из того, что речь идет о модели, и принцип неувеличения числа сущностей сверх необходимости (бритва Оккама) побуждает нас избегать ненужного дублирования одних и тех же характеристик как моделируемого явления, так и его связей с внешним миром. Что касается M и E , словами «множество различных реализаций» подчеркивается именно уникальность методов в этих множествах. На самом деле, язык математики хорош именно тем, что иногда позволяет одними и теми же соотношениями записать различные законы, имеющие место в совсем различных предметных областях. Излагаемый здесь подход позволяет учитывать этот факт и пользоваться им. Однако для этого необходимо знать, сколько и каких различных функциональных зависимостей имеется в нашем распоряжении. Множества M и E как раз и являются такими «хранилищами» различных функциональных зависимостей для модели-компоненты.

Далее будут упоминаться процессы модели-компоненты. Подробное объяснение того, что такое процессы будет дано далее, пока же скажем, что содержательно процессы подобны, например, системным службам операционной системы компьютера. Они работают всегда и при этом одновременно. Будем считать, что число процессов модели – N . Каждый процесс p_j , $j=1,\dots,N$, последовательно осуществляет некоторый конечный набор возможных для него элементарных действий M_j , который будем называть множеством его методов-элементов, возможно, в зависимости от возникающих в системе ситуаций E_j , на которые процесс умеет реагировать, их будем называть множеством его методов-событий.

Этим исчерпывается описание основных базисных множеств.

Вспомогательных базисных множеств нет.

Перейдем теперь к соотношениям типизации и аксиомам. Уже упоминавшееся соотношение $x \subset X$ определяет внутренние характеристики модели как подмножество всех ее характеристик.

$$\{m_{j,real} \subset M_j \times M\}_{j=1}^N \quad (3)$$

Для каждого процесса p_j задается отображение множества его методов-элементов M_j во множество их реализаций M . Таким образом, для любого $m \in M_j$ однозначно определяется его реализация $\tilde{m} \in M$. Формально это можно записать как аксиомы

$$R_3 : (\forall m \in M_j)(\exists ! \tilde{m} \in M)(\{m, \tilde{m}\} \in m_{j,real})_{j=1}^N.$$

Разница между имеющими одинаковую реализацию методами, с точки зрения процесса, может быть, например, в способе коммутации их параметров с характеристиками модели, о чем речь пойдет ниже. Не запрещаются (хотя и не рекомендуются) и просто «синонимы».

$$\{e_{j,real} \subset E_j \times E\}_{j=1}^N \quad (4)$$

Для каждого процесса p_j задается отображение множества его методов-событий E_j во множество их реализаций E . Для любого $e \in E_j$ однозначно определяется его реализация $\tilde{e} \in E$. Аксиомы

$$R_4 : (\forall e \in E_j)(\exists ! \tilde{e} \in E)(\{e, \tilde{e}\} \in e_{j,real})_{j=1}^N$$

утверждают этот факт. Так же, как и в случае методов-элементов, различные с точки зрения процесса события, могут иметь одинаковые реализации.

$$\{m_{j,in} \subset M_j \times \beta(X)\}_{j=1}^N \quad (5)$$

Коммутируются внутренние и внешние характеристики модели с входящими параметрами методов-элементов j -го процесса. Включение должно определить, какое подмножество характеристик модели передается каждому из методов-элементов. Это можно выразить аксиомами

$$R_5: \left\{ \left(\forall m \in M_j \right) \left(\exists ! r \in \beta(X) \right) \left(\{m, r\} \in m_{j,in} \right) \right\}_{j=1}^N.$$

$$\{m_{j,out} \subset M_j \times \beta(X)\}_{j=1}^N \quad (6)$$

Коммутируются возвращаемые параметры методов-элементов с внутренними характеристиками модели. При этом должны выполняться аксиомы

$$R_6: \left\{ \left(\forall m \in M_j \right) \left(\exists ! r \in \beta(x) \right) \left(\{m, r\} \in m_{j,out} \right) \right\}_{j=1}^N.$$

$$\{e_{j,in} \subset E_j \times \beta(X)\}_{j=1}^N \quad (7)$$

Коммутируются внутренние и внешние характеристики модели с входящими параметрами методов-событий j -го процесса. Включение должно определить, какое подмножество характеристик модели передается каждому из методов-событий. При этом должны выполняться аксиомы

$$R_7: \left\{ \left(\forall e \in E_j \right) \left(\exists ! r \in \beta(X) \right) \left(\{e, r\} \in e_{j,in} \right) \right\}_{j=1}^N.$$

Соотношениями

$$\{sw_j \subset E_j \times M_j \times M_j\}_{j=1}^N \quad (8)$$

определяются переключения методов-элементов в каждом процессе. Если при определенных условиях возможно пере-

ключение процесса с выполнения метода $A \in M_j$ на выполнение метода $B \in M_j$ (а таким условием в силу гипотезы о замкнутости может быть лишь некое сочетание внутренних и внешних характеристик модели $\{x, a\} \in X$, – в виртуальном мире модели просто ничего больше нет), то должен быть создан единственный метод-событие $e_{AB}(x, a) \in E_j$, прогнозирующий и вычисляющий такое переключение. Возможны также события вида $e_{AA}(x, a)$, не переключающие метод-элемент $A \in M_j$ на другой, а лишь прерывающие его выполнение (например, для синхронизации результатов его вычислений с другими процессами модели). Таким образом, время и порядок переключений элементов каждого процесса определяется тем, что происходит внутри и вне модели. Вообще говоря, в самых простых процессах, событий может и не быть, тогда не будет и переключений. Если же множество событий j -го процесса – E_j , непусто, то соотношение типизации (8) должно удовлетворять следующим аксиомам

$$R_8 : \left(\left(\forall e \in E_j \right) \left(\exists ! r \in M_j \times M_j \right) \left(\{e, r\} \in sw_j \right) \right) \& \\ \& \left(\left(\{e, r\} \in sw_j, \{\tilde{e}, \tilde{r}\} \in sw_j, r = \tilde{r} \right) \Rightarrow \left(e = \tilde{e} \right) \right)_{j=1}^N$$

утверждающим, что отображения множеств методов-событий во множества переключений биективны.

Соотношение

$$\left\{ m_j^0 \subset M_j \right\}_{j=1}^N \quad (9)$$

определяет начальные методы-элементы процессов.

Соотношения типизации

$$\left\{ p_j \subset \beta(M_j) \times \beta(E_j) \times M_j \times \beta(E_j \times M_j \times M) \right\}_{j=1}^N \quad (10)$$

определяют процессы модели, а соответствующий им набор аксиом

$$R_9: \left\{ p_j = \left\{ M_j, E_j, m_j^0, sw_j \right\} \right\}_{j=1}^N,$$

уточняет это понятие. Процесс – это множества элементов и событий, начальный элемент и правила переключений. Множества M_j и E_j – это множества элементов и событий j -го процесса; m_j^0 – его начальный элемент; sw_j – правила переключения элементов, которые были подробно описаны выше. Формальное определение семейства родов структур «модель-компонент» на этом полностью заканчивается.

Тем не менее, дополним набор аксиом семейства родов структур «модель-компонента» еще двумя. Эти аксиомы в некотором смысле стоят особняком, поскольку ничего не добавляют к формальному определению отношений, задаваемых на базисных множествах рассматриваемого семейства родов структур – упомянутые отношения уже полностью определены. Смысл дополнительных аксиом R_{10} и R_{11} в том, чтобы описать, что и как мы собираемся делать дальше с математическими объектами семейства родов структур «модель-компонента».

Аксиома R_{10} требует однозначности вычисления характеристик модели-компоненты. Возможность однозначности вычисления характеристик постулируется гипотезой о замкнутости и обсуждалась выше. Для модели-компоненты это требование означает, например, что два метода-элемента разных процессов не должны одновременно менять одну и ту же характеристику. А гипотеза о замкнутости утверждает, что этого возможно добиться. Собственно, для модели-компоненты этим все и ограничивается, и можно было бы не выделять это требование в аксиому. Однако, при объединении компонент в

комплекс, все оказывается не так просто, и эта аксиома начинает работать.

Аксиома поведения R_{11} задает следующие правила запуска на счет любого математического объекта семейства родов структур «модель-компоненты».

Во-первых, считается, что в начале шага моделирования известны текущие элементы всех процессов и все внутренние характеристики модели (на первом шаге – это начальные значения внутренних характеристик и начальные элементы процессов).

Во-вторых, предполагается, что внешние характеристики модели возможно определить в любой момент модельного времени.

Далее

1. Вычисляются события связанные с текущими элементами процессов. Связь событий с текущими элементами процессов определяется правилами переключений (8). Вычисляться события могут параллельно, однако для продвижения вычислительного процесса далее, следует дождаться завершения вычислений всех событий. Если есть наступившие события, проверяется, нет ли переходов к быстрым элементам из множеств $\{f_j\}_{j=1}^N$, если они есть – выполняются соответствующие быстрые элементы (они становятся текущими). Вычисляться они могут также параллельно, однако для продвижения вычислительного процесса далее, следует дождаться завершения вычислений всех быстрых элементов, затем возврат к началу п.1; если нет переходов к быстрым элементам – совершаются переходы к новым медленным элементам из множеств $\{s_j\}_{j=1}^N$, затем возврат к началу п.1.
2. Если нет наступивших событий – из всех прогнозов событий выбирается ближайший $\Delta\tau$.

- Если стандартный шаг моделирования Δt не превосходит прогнозируемого времени до ближайшего события, $\Delta t \leq \Delta\tau$ – вычисляем текущие медленные элементы со стандартным шагом Δt . В противном случае вычисляем их с шагом времени до ближайшего спрогнозированного события $\Delta\tau$. Медленные элементы из множества $\{s_j\}_{j=1}^N$, также можно вычислять параллельно, с ожиданием завершения последнего.

4. Возвращаемся к началу п.1.

Модели-компоненты могут объединяться в **модель-комплекс**, при этом (необязательно) может оказаться, что некоторые компоненты явно моделируют внешние переменные некоторых других компонент. Для того чтобы полностью описать комплекс, достаточно указать:

- Какие компоненты и в каком количестве экземпляров в него входят.
- Коммутацию компонент внутри комплекса, если она имеет место, т. е., какие внутренние переменные каких компонент являются какими внешними переменными и каких именно компонент комплекса.

Комплекс, состоящий из многих компонент, вовне может проявляться в качестве единой компоненты. Сформулируем основной теоретический результат работы:

Предложение 2

Пусть имеются n математических объектов семейства родов структур $\Sigma_{N_1}, \dots, \Sigma_{N_n}$ «модель-компонента» с базисными множествами

$$X_1, M_1, E_1, \{M_{1,j}\}_{j=1}^{N_1}, \{E_{1,j}\}_{j=1}^{N_1}, \dots, X_n, M_n, E_n, \{M_{n,j}\}_{j=1}^{N_n}, \{E_{n,j}\}_{j=1}^{N_n}.$$

Тогда возможно распространение рода структуры $\Sigma_{N_1 + \dots + N_n}$ семейства «модель-компонента» на базисные множества модели-комплекса, составленной из этих моделей-компонент:

$$X_1 \cup \dots \cup X_n, M_1 \cup \dots \cup M_n, E_1 \cup \dots \cup E_n, \{M_j\}_{j=1}^{N_1 + \dots + N_n}, \{E_j\}_{j=1}^{N_1 + \dots + N_n},$$

при этом:

1. В объединение характеристик $X_1 \cup \dots \cup X_n$ входят все характеристики компонент, т.е., формально различными считаются даже одинаковые в предметной области характеристики разных компонент. Объединения реализаций методов-элементов $M_1 \cup \dots \cup M_n$ и методов-событий $E_1 \cup \dots \cup E_n$ понимаются в обычном смысле.
 2. Если множества внутренних характеристик моделей-компонент $x_1 \subset X_1, \dots, x_n \subset X_n$ имеют непустые попарные пересечения, то исходный набор из n моделей-компонент придется пополнить еще некоторым количеством L объектов семейства родов структур «модель-компоненты». Первоначальное множество характеристик комплекса также пополняется одной новой характеристикой на каждую дополнительную компоненту.
 3. Если попарно пересекаются внутренние и внешние характеристики компонент: $x_i \subset X_i, a_j \subset X_j, i \neq j$ и $x_i \cap a_j \neq \emptyset$, то эти характеристики коммутируются, при этом коммутируемые внешние характеристики компонент исключаются из набора внешних характеристик комплекса, так как становятся его внутренними характеристиками.
 4. Если попарно пересекаются внешние характеристики моделей-компонент – они либо отождествляются, либо, как и в случае с внутренними характеристиками, неоднозначность разрешается путем добавления в комплекс некоторого количества дополнительных компонент и характеристик.
-

Таким образом, модели-компоненты допускают объединение в комплексы с возможной (но не всегда обязательной) коммутацией некоторых внутренних характеристик некоторых компонент с некоторыми внешними характеристиками других, воз-

можным дополнением первоначального набора компонент некоторыми новыми и коррекцией части соответствий в соотношениях (5) – (7) для некоторых компонент. Как было показано, полученная таким образом модель-комплекс обладает структурой рода «модель-компонент» (быть может, после добавления в комплекс еще некоторого количества компонент, разрешающих возникшие при объединении базисных множеств неоднозначности), и, стало быть, в свою очередь может входить в новые модели-комплексы.

Этим способом можно строить сколь угодно сложные фрактальные конструкции комплексов, состоящих из компонент, которые сами затем становятся компонентами комплексов. При этом, тем не менее, все вычислительные процессы конструкций произвольной сложности, полностью определяются аксиомой поведения R_{11} семейства родов структур «модель-компонент».

Данное свойство замкнутости семейства моделей-компонент относительно операции объединения их в модели-комплексы, позволяет предложить новую парадигму программирования для реализации имитационных моделей сложных систем – модельно-ориентированное программирование. Основной его концепцией и рабочей единицей является понятие модели-компоненты, а собственно программирование заменяется декларативными описаниями моделей-компонент, их объединений в модели-комплексы, а также написанием программ, реализующих методы-элементы и методы-события, которые в силу гипотезы о замкнутости могут быть выполнены в функциональной парадигме. Таким образом, модельно-ориентированное программирование исключает наиболее трудоемкое в написании и отладке императивное программирование.

Далее в работе сравниваются основные понятия и методы модельного синтеза и модельно-ориентированного программирования с основными понятиями и методами объектного анализа и объектно-ориентированного программирования.

Такие понятия как класс, объект, типизация в двух рассматриваемых подходах понимаются примерно одинаково. Следует лишь заметить, что корни этих понятий следует искать конечно не в C++ и даже не в Симуле-67, а скорее в работах Н. Бурбаки, структуралистов XX века, вплоть до Эрлангенской программы Ф. Клейна.

Примерно одинаково понимаются также характеристики и методы. Что же касается использования методов – здесь имеется существенная разница. Она состоит в том, что в объектном программировании объект затем и нужен, чтобы вызывать его методы в различных программах, причем в качестве параметров методу можно передать, а также принять от него любые переменные, удовлетворяющие его сигнатуру – не обязательно характеристики объекта. В модельно-ориентированном программировании метод модели-компоненты может работать лишь с ее характеристиками, и вызывать его «вручную» нет ни возможности, ни необходимости – он будет вызван автоматически, когда это потребуется. Инкапсуляция в модельно-ориентированном программировании такова, что не позволяет напрямую добраться до методов. Оперировать можно лишь с моделями-компонентами.

Это вовсе не означает, что методы совсем недоступны. Наоборот, например, в реализованном макете распределенной системы имитационного моделирования, библиотеки методов публикуются в Интернете для общего использования и возможны модели-компоненты, не имеющие ни одного локального метода, притом все реализации методов могут физически находиться в разных местах Сети. Тем не менее, воспользоваться методом можно лишь включив его в состав некоторой модели-компоненты – таков уровень инкапсуляции. В некоторых случаях это может показаться усложнением, но оно оплачено отсутствием забот об организации поведения модели-компоненты – она всегда ведет себя так, как умеет, и поэтому

включение ее в любой комплекс всего лишь вопрос правильной коммутации.

Скажем несколько слов о наследовании. Отношение наследования для множества классов объектно-ориентированного языка программирования есть отношение частичного порядка. Классы, не имеющие предков, но обладающие потомками, называются по отношению к ним корневыми или базовыми. Классы, не имеющие потомков, называются листовыми.

Проектирование в объектно-ориентированной парадигме большой программной системы состоит в воплощении базовых понятий и представлений этой системы в базовые классы объектов и построении затем иерархии классов, развивающих, конкретизирующих и воплощающих эти идеи во множестве листовых классов, с помощью которых и будет строиться целевая программная система. Такой дедуктивный способ проектирования большой программной системы хорош, когда она творится «с чистого листа», как мир у Платона. В имитационном моделировании, однако, гораздо чаще возникает задача не создания новых миров, а воспроизведения фрагментов уже существующего. В подобном фрагменте запросто могут быть собраны вместе «...диван, чемодан, саквойж, картина, корзина, картонка и маленькая собачонка». Друг из друга они не выводятся, а восходят вверх до их общих предков – достаточно бессмысленно. Для подобных задач «базовому» варианту объектного подхода недостает наследования снизу вверх (в UML такое наследование имеется). В модельно-ориентированном программировании объединение моделей-компонент в модель-комплекс можно считать множественным наследованием снизу вверх.

Даже если в объектно-ориентированном проекте с помощью наследования построена самая замечательная иерархия классов, все равно все сложности организации вычислительного процесса, состоящего в использовании разработанных и отлаженных методов листовых классов, лежит на разработчике си-

стемы: чтобы система что-то делала – необходимо организовать вызов нужных методов в нужной последовательности. Самый сложный этап построения системы остается неформализованным – это искусство.

Попытки формализовать процесс проектирования сложных программных систем породили язык UML. По-видимому, и в самом деле на UML можно описать любую систему, и даже с нескольких точек зрения. Вопрос в том, что делать дальше с таким описанием – здесь нет единства мнений. Некоторые специалисты считают, что основная ценность UML как раз в применении как средства документирования и обмена формализованными описаниями на стадиях эскиза и проектирования сложных систем. Тем не менее, имеется и ряд средств, позволяющих компилировать UML-описания в заготовки классов универсальных языков программирования, и в этом случае можно говорить о режиме использования UML в качестве языка программирования. Однако здесь мы снова остаемся в рамках объектной парадигмы – получаем иерархию классов и заготовок классов, но не избавляемся от необходимости писать императивные программы, вызывающие в нужном порядке методы этих классов.

Далее описывается программная реализация макета рабочей станции пиинговой сети имитационного моделирования в Интернете.

Предлагаемая инструментальная система во-первых, помогает построить синтез сложной многокомпонентной модели, что само по себе обычно является нетривиальной задачей, и, во-вторых, позволяет создать в Интернете пиинговую сеть моделирования, в которой с одной стороны, каждый участник сети может предоставлять во всеобщий доступ (опубликовать) разработанные им методы, а с другой стороны, может использовать в разрабатываемых им моделях подходящие методы, разработанные другими участниками и опубликованные ими в Интернете. При этом от публикуемых методов требуется

лишь, чтобы они по своему текущему состоянию (набору внутренних характеристик), набору внешних характеристик (параметров) и интервалу времени моделирования, могли вычислять внутренние переменные в конце этого интервала. Как правило, так или иначе именно этим и занимается большинство компьютерных программ. Публикация метода, помимо описания его назначения и интерфейса на естественном языке, предполагает в дальнейшем возможность его удаленных вызовов. В основе инструментальной системы лежит предложенная в работе концепция анализа и синтеза сложных систем, ориентированная на параллельное выполнение методов, одновременно протекающих в модельном времени.

Основой сети распределенного моделирования является программное обеспечение пиинговой рабочей станции, оно состоит из клиентской и серверной частей. Клиентская часть ответственна за создание моделей. Она содержит средства, позволяющие по описаниям на специальном формальном непрограммном языке ЯОКК (языке описания комплексов и компонент) методов, событий, компонент и комплексов строить информационную структуру модели, в первую очередь ее базу данных, а также средства поддержки выполнения модели и наблюдения за результатами моделирования. При этом, методы, обеспечивающие функциональность модели могут быть как локальными, так и удаленными. Серверная часть пиингового клиента предоставляет в первую очередь сервис удаленных вызовов опубликованных на этом хосте методов, а также сервис просмотра каталога и описаний этих методов.

Архитектура программного обеспечения рабочей станции такова, что вся модель от начала и до конца всегда собирается на одной рабочей станции. При этом, вообще говоря, ни один из методов модели, будь то элемент или же событие, может не быть реализован на этой рабочей станции. Любые методы модели могут быть «чужими» и выполняться удаленно на других компьютерах сети распределенного моделирования. В этом

контексте слова о том, что вся модель находится на рабочей станции, означают, что на ней по исходным описаниям создается база данных модели, которая затем заполняется начальными данными, а впоследствии содержит все характеристики модели, как внутренние так и внешние, на всех отработавших шагах моделирования. Распределено вызываются только методы, которым в момент вызова передаются их входные параметры, а после вызова принимаются их возвращаемые параметры.

Для формального описания компонент и комплексов модели, служит специальный непроцедурный язык описания комплексов и компонент (**ЯОКК**).

В **Заключении** подводятся итоги работы.

В **Приложении** приведены примеры применения предлагаемых в работе методов модельного синтеза и модельно-ориентированного программирования при создании имитационных моделей ряда сложных систем.

Приводятся две модели, разработанные в отделе «Имитационные системы» ВЦ АН СССР, и ВЦ РАН в конце 80-х, начале 90-х гг. и воспроизводящие различные сценарии ведения космических войн в рамках Стратегической оборонной инициативы Р. Рейгана.

Приводится распределенная модель взаимодействия трех стран, выполненная на основе разработанной в отделе «Имитационные системы и исследование операций» ВЦ РАН в 2001 – 2014 гг. Экономико-демографо-экологической модели (**ЭДЭМ**).

Упомянутые выше три модели – результат коллективной работы сотрудников отдела и их коллег из дружественных организаций. Вклад автора – организация вычислительного процесса имитации и системная интеграция.

Наконец, приводятся две реализованные автором тестовые модели для отладки макета рабочей станции сети распределенного имитационного моделирования – модель «Пешеходы и му-

ха» (муха фон Неймана) и модель простейшего бизнес-процесса «Передел рынка».

ОСНОВНЫЕ ВЫВОДЫ И РЕЗУЛЬТАТЫ РАБОТЫ

- Разработана концепция модельного синтеза и модельно-ориентированного программирования, позволяющая решать задачи описания, синтеза и реализации имитационных моделей сложных многокомпонентных систем. Предлагаемая технология позволяет исключить из проекта создания имитационной модели императивное программирование и получить исполняемый код высокой степени параллельности.
- Разработан язык описания комплексов и компонент (ЯОКК), предназначенный для описания моделей-компонент и синтеза из них моделей-комплексов.
- На основании формализации понятия модели сложной системы разработана инструментальная система распределенного имитационного моделирования, объединяющая декларативную систему программирования на базе языка ЯОКК, базу данных, где хранятся модели и их данные, и систему поддержки выполнения имитационного эксперимента.
- Разработана архитектура пиинговой сети имитационного моделирования в Интернете, отличающаяся от известных, и позволяющая разработчику полностью контролировать модель, даже если все ее составляющие – удаленные.
- Реализовано программное обеспечение макета рабочей станции пиинговой сети имитационного моделирования, на котором проверены основные моменты разработанной концепции моделирования.

СПИСОК ПУБЛИКАЦИЙ ПО ТЕМЕ ДИССЕРТАЦИИ

Публикации в журналах из списка ВАК

1. Бродский Ю.И. Роды структур Н. Бурбаки в задаче синтеза имитационных моделей сложных систем и модельно-

ориентированное программирование //Журнал Вычислительной математики и математической физики, 2015, том 55, № 1, с. 153-164.

2. *Brodsky Yu.I. Bourbaki's Structure Theory in the Problem of Complex Systems Simulation Models Synthesis and Model-Oriented Programming* //Computational Mathematics and Mathematical Physics Vol. 55 No. 1, 2015. P. 148-159.
3. *Бродский Ю.И. О модельном синтезе и модельно-ориентированном программировании* //Научное обозрение №12, 2014. С. 151-157.
4. *Мягков А.Н., Бродский Ю.И. Об управлении временем в распределённых имитационных моделях* //ТРУДЫ МФТИ, Т. 4, № 3(15), 2012, С. 181-186.
5. *Бродский Ю.И., Павловский Ю.Н. Разработка инструментальной системы распределенного имитационного моделирования.* //Информационные технологии и вычислительные системы, №4, 2009, С. 9-21.
6. *Бродский Ю.И., Павловский Ю.Н. Разработка инструментальной системы распределенного имитационного моделирования* //Прикладные проблемы управления макросистемами. Под ред. Попкова Ю.С., Путилова В.А. Труды ИСА РАН , Т.39, М.: ИСА РАН, 2008. С. 79-99.
7. *Бродский Ю.И., Новицкий В.И., Павловский Ю.Н. Об алгоритме, формирующем иерархическую систему инвариантов изоморфизмов отображений конечных множеств в себя* //Дискретный анализ и исследование операций, Серия 2, Том 13, №2, 2006, С. 21-30.
8. *Тарасова Н.П., Павловский Ю.Н., Кручинина Е.Б., Белотелов Н.В., Бродский Ю.И., Оленев Н.Н. Методика применения эколого-демографо-экономической модели виртуального государства для оценки качества человеческого потенциала* //Менеджмент в России и за рубежом, №4, 2006, С.38-45.
9. *Бродский Ю.И., Гринберг Я.Р., Павловский Ю.Н. Имитационное моделирование в распределенной информационно-*

- вычислительной среде //Проблемы вычислений в распределенной среде: прикладные задачи. Ч.2, Труды ИСА РАН. - М.: РОХОС, 2004, С. 95-110.
10. Бродский Ю.И., Гринберг Я.Р., Павловский Ю.Н. Имитационное моделирование в распределенной информационно-вычислительной среде //Проблемы вычислений в распределенной среде Труды ИСА РАН. - М.: РОХОС, 2003, С. 121-140.
11. Белотелов Н.В., Бродский Ю.И., Оленев Н.Н., Павловский Ю.Н. Эколого-социально-экономическая модель: гуманистичный и информационный аспекты //Информационное общество. № 6. 2001. С. 43-51.

Компьютерные программы, зарегистрированные в РОСПАТЕНТе

12. Бродский Ю.И. Программное обеспечение для макета рабочей станции пионерской сети распределенного имитационного моделирования «Прототип-Д» //Свидетельство № 2015613336 о государственной регистрации от 12.03.2015.
13. Бродский Ю.И. Анализатор типов данных для компилятора с языка описаний комплексов и компонент (ЯОКК) //Свидетельство № 2015613334 о государственной регистрации от 12.03.2015.
14. Белотелов Н.В., Бродский Ю.И., Оленев Н.Н., Павловский Ю.Н. Имитационная экономико-демографо-экологическая модель (ЭДЭМ) динамики государства //Свидетельство № 2015613120 о государственной регистрации от 05.03.2015.
15. Белотелов Н.В., Бродский Ю.И., Мягков А.Н., Оленев Н.Н., Павловский Ю.Н. Распределенная имитационная модель взаимодействия нескольких государств с учетом экономических, демографических и экологических процессов //Свидетельство № 2015613335 о государственной регистрации от 12.03.2015.

Монографии и главы в монографиях

16. *Бродский Ю.И.* Модельный синтез и модельно-ориентированное программирование М.: ВЦ РАН, 2013, 142 с.
17. *Бродский Ю.И.* РАСПРЕДЕЛЕННОЕ ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ: эвристика, теория, инструментальные средства – Saarbrücken: LAP LAMBERT Academic Publishing, 2011, 152 с.
18. *Бродский Ю.И.* Распределенное имитационное моделирование сложных систем М.: ВЦ РАН, 2010, 156 с.
19. *Белотелов Н.В., Бродский Ю.И., Павловский Ю.Н.* Глава 16 «Имитационное математическое моделирование и прогноз глобальных процессов» // В монографии «Прогноз и моделирование кризисов и мировой динамики» /под ред. А.А. Акаева, А.В. Коротаева, Г.Г. Малинецкого, М.: Изд-во ЛКИ, 2010, 352 с.
20. *Белотелов Н.В., Бродский Ю.И., Павловский Ю.Н.* Сложность. Математическое моделирование. Гуманитарный анализ. М.: Книжный дом «ЛИБРОКОМ», 2009, 320 с.
21. *Белотелов Н.В., Бродский Ю.И., Павловский Ю.Н.* Компьютерное моделирование демографических, миграционных, эколого-экономических процессов средствами распределенных вычислений. М.: ВЦ РАН, 2008. 123 с.
22. *Павловский Ю.Н., Белотелов Н.В., Бродский, Ю.И. Оленев Н.Н.* Опыт имитационного моделирования при анализе социально-экономических явлений М.: МЗ Пресс, 2005, 137 с.
23. *Белотелов Н.В., Бродский, Ю.И. Оленев Н.Н., Павловский Ю.Н., Тарасова Н.П.* Проблема устойчивого развития: естественно-научный и гуманитарный анализ. М.: Фазис. 2004. 108 с.
24. *Бродский Ю.И., Лебедев В.Ю.* Инструментальная система имитации MISS М.: ВЦ АН СССР, 1991, 180 с.

Публикации в учебных и научно-справочных изданиях

25. Павловский Ю.Н., Белотелов Н.В., Бродский Ю.И. Компьютерное моделирование. Учеб. пособие для вузов. М.: Физматкнига, 2014. 304 с.
26. Brodsky Yury I. Simulation Software //System Analysis and Modeling of Integrated World Systems – Volume 1, Oxford: EOLSS Publishers Co. Ltd., 2009, P. 287-298.
27. Brodsky Yury I., Tokarev Vladislav V. Fundamentals of simulation for complex systems. //System Analysis and Modeling of Integrated World Systems – Volume 1, Oxford: EOLSS Publishers Co. Ltd., 2009, P. 235-250.
28. Белотелов Н.В., Бродский Ю.И., Винокуров С.Ф., Высоцкий М.Н., Коровко М.А., Кручинина Е.Б., Миносьянц С.С., Мягков, А.Н., Оленев Н.Н., Павловский Ю.Н., Тарасова Н.П. Лабораторный практикум по математическому моделированию. М.: РХТУ им. Д.И. Менделеева. 2009. 80 с.
29. Павловский Ю.Н., Белотелов Н.В., Бродский Ю.И. Имитационное моделирование: учеб. пособие для студ. высш. учеб. заведений М.: Изд. центр «Академия», 2008, 236 с.
30. Белотелов Н.В., Бродский Ю.И., Кручинина Е.Б., Оленев Н.Н., Павловский Ю.Н. Имитационная игра на основе эколого – демографо – экономической модели (ЭДЭМ): описание и инструкция пользователю (учебное пособие) М.: РХТУ им. Д.И. Менделеева, 2003. 84 с.

Публикации в других изданиях

31. Brodsky Yu.I. Model synthesis and model-oriented programming – the technology of design and implementation of simulation models of complex multicomponent systems //In the World of Scientific Discoveries, Series B, 2014, Vol.2, No 1, P. 12-31.
32. Brodsky Yu.I. Bourbaki's structure theory in the problem of multi-component simulation models synthesis //Innovative Information Technologies, Part 2, M.: HSE, 2014, P. 234-244.
33. Бродский Ю.И. О приближенной декомпозиции модели-компоненты //Моделирование, декомпозиция и оптимизация

- ция сложных динамических процессов, 2014. Т.29, №1(29), С. 119-127.
34. *Бродский Ю.И.* О модельном синтезе и модельно-ориентированном программировании //Гений В.Г. Шухова и современная эпоха: международный конгресс (Москва, 17–18 апреля 2014 г.): материалы международного конгресса /под ред. А.Г. Багдасарьян, Е.А. Гаврилиной, М.: МГТУ им. Н.Э. Баумана, 2014. С. 190-193.
35. *Brodsky Yu.I.* Model Synthesis and Model-Oriented Programming //Third International Conference High Performance Computing HPC-UA 2013: Collection of scientific papers, October 7-11, 2013, Kyiv, Р. 69-76.
36. *Бродский Ю.И.* О концепции декларативно-поведенческого программирования в имитационном моделировании сложных распределенных систем //Advanced Science, №1(2), 2013, ВятГУ, г. Киров, С. 41-73.
37. *Бродский Ю.И.* Анализ и синтез имитационных моделей сложных систем //Моделирование, декомпозиция и оптимизация сложных динамических процессов, 2013. Т. 28, №1(28), С. 108-122.
38. *Бродский Ю.И.* Модельный синтез и модельно-ориентированное программирование, как технология реализации имитационных моделей сложных многокомпонентных систем, с ориентацией на параллельные и распределенные вычисления //Материалы конференции «Имитационное моделирование. Теория и практика». ИММОД-2013. – Казань: Изд-во «Фэн» Академии наук РТ, 2013. – Т1, С. 114-118.
39. *Бродский Ю.И., Мягков А.Н.* Декларативное и императивное программирование в имитационном моделировании сложных многокомпонентных систем //Инженерный журнал: наука и инновации. 2012. № 2 (2). С. 33.

40. *Бродский Ю.И., Мягков А.Н.* Анализ деловой игры «взаимодействие 3-х стран» //Инженерный журнал: наука и инновации. 2012. № 2 (2). С. 32.
41. *Бродский Ю.И.* Методы декомпозиции в программировании имитационных моделей сложных систем //Моделирование, декомпозиция и оптимизация сложных динамических процессов, 2012. Т. 27, №1(27), С. 53-78.
42. *Бродский Ю.И.* Некоторые вопросы синтеза сложных распределенных имитационных моделей //Моделирование, декомпозиция и оптимизация сложных динамических процессов, 2010. Т. 25, №1(25), С. 25-40.
43. *Белотелов Н.В., Бродский Ю.И., Павловский Ю.Н.* Разработка инструментальной системы распределенного моделирования. //IV Всероссийская научная конференция «Математическое моделирование развивающейся экономики и экологии», ЭКОМОД-2009, /Сборник трудов. Киров, изд-во. ВятГУ, 2009, С. 61-73.
44. *Бродский Ю.И.* Описание, компоновка и работа модели в инструментальной системе распределенного моделирования. //Моделирование, декомпозиция и оптимизация сложных динамических процессов, 2008. Т. 23, №1(23), С. 24-46.
45. *Бродский Ю.И.* К разработке концепции построения инструментальной системы распределенного моделирования. //Моделирование, декомпозиция и оптимизация сложных динамических процессов, 2007. Т. 22, №1(22), С. 14-34.
46. *Бродский Ю.И.* О переходе от макроописания математической модели к ее объектно-событийному представлению //Моделирование, декомпозиция и оптимизация сложных динамических процессов, 2004. Т. 19, №1(19), С. 22-28.
47. *Бродский Ю.И.* Эколого-социально-экономическая имитационная модель: технология реализации. //Моделирование, декомпозиция и оптимизация сложных динамических процессов, 2001. Т. 16, №1(16), С. 89-107.

48. *Бродский Ю.И.* Разработка инструментальных средств имитационного моделирования сложных технических комплексов для параллельных вычислительных систем. //Моделирование, декомпозиция и оптимизация сложных динамических процессов, 1999. Т. 14, №1(14), С. 33-92.
49. *Бродский Ю.И.* Проблемы создания центра имитационного моделирования в Интернет //Моделирование, декомпозиция и оптимизация сложных динамических процессов. 1998. Т. 13, №1(13), С. 29-35.
50. *Бродский Ю.И., Лебедев В.Ю.* Инструментальная система для построения имитационных моделей хорошо структурированных организационно-технических комплексов //Вопросы кибернетики. Проблемы математического моделирования и экспертные системы, М.: НС АН СССР по комплексной проблеме «Кибернетика», 1990, С. 49-64.

Бродский Юрий Игоревич

Проблема описания и синтеза распределенных имитационных моделей
сложных многокомпонентных систем

Подписано в печать 28.05.2015

Формат бумаги 60×84 1/16

Уч.-изд. л. 2. Усл.-печ. л. 2,75

Тираж 100 экз. Заказ 12

Отпечатано на ротапринтах

в Федеральном государственном бюджетном учреждении науки
Вычислительный центр им. А.А.Дородницына Российской академии наук
119333, Москва, ул. Вавилова, 40