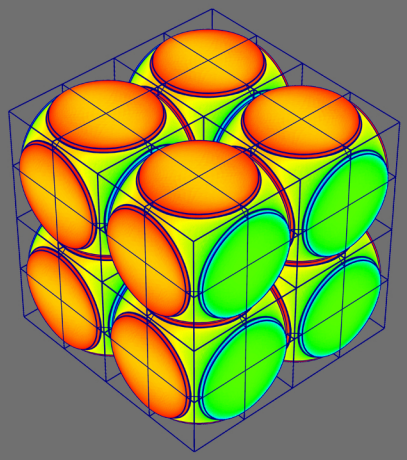
ISSN 2309-3684



Математическое моделирование и численные методы

Бродский Ю.И. Структурная теория сложных систем. Модельный синтез. Математическое моделирование и численные методы, 2022, № 3, с. 98–123.

Источник: https://mmcm.bmstu.ru/articles/290/

Параметры загрузки:
IP: 216.73.217.61
09.11.2026 04:17:26

УДК 519.87 DOI: 10.18698/2309-3684-2022-3-98123

Структурная теория сложных систем. Модельный синтез

© Ю.И. Бродский

ФИЦ ИУ РАН, Москва, 119333, Россия

В данной статье прежде всего хотелось упорядочить результаты работ автора последних двух десятков лет в области структурной теории моделирования сложных систем и практики реализации таких систем с единых позиций. На основе гуманитарного анализа ключевых свойств сложных систем, признаваемых таковыми рядом авторитетных исследователей и практиков этой области, и предположения о возможности построения математической компьютерной модели сложной системы, — гипотезы о замкнутости, — предлагается формальное определение компьютерной модели сложной системы, как рода структуры в смысле Н. Бурбаки — род структуры М (модель). Класс математических объектов, определяемый родом структуры М обладает следующими двумя свойствами: комплекс, созданный объединением конечного числа математических объектов рода структуры М по определенным правилам, сам является объектом этого рода структуры. Организация вычислительного процесса для всех математических объектов рода структуры М однотипна и поэтому может быть реализована единой универсальной программой организации имитационных вычислений. Наличие этих двух свойств у представителей рода структуры М позволяет построить сквозную технологию описания, синтеза и программной реализации моделей сложных систем — Модельный синтез и Модельно-ориентированное программирование. Изучая морфизмы базисных множеств построенной с помощью модельного синтеза модели рода структуры М, и инварианты, ограничивающие такие морфизм, мы получаем формальный математический язык исследования сложных открытых (меняющих свой состав) систем. Ведя традиционный по форме гуманитарный дискурс, можно все время соотносить его с соответствующим объектом рода структуры М — транслируя на математический язык гуманитарные понятия языка более высокого уровня. Предлагаемая теория имеет практическое применение в области разработки, описания и реализации сложных программных систем. Предлагается новая программистская парадигма — Модельно-ориентированное программирование, являющееся полной реализацией методов САПР в программировании. При разработке программной системы удается оставаться в рамках декларативного программирования, избегая императивного, что существенно упрощает как ее разработку и реализацию, так и последующую отладку.

Ключевые слова: *сложные системы, структурная теория, модельный синтез, геометрическая теория поведения, модельно-ориентированное программирование, формализация описательных наук*

Введение. Из диалога Платона «Парменид» [1] мы знаем, что за четыреста с лишним лет до Нашей эры, Афины посетили маститый элейский софист Парменид со своим любимым учеником Зеноном, знаменитым своими апориями. Многие из афинских любителей мудрости искали встречи и бесед с ними, в том числе и молодой

Сократ и совсем юный Аристотель, но не знаменитый философ, который к тому времени еще не успел родиться, а его тезка, впоследствии политик, один из будущих тридцати тиранов, которых Спарта, победив в Пелопонесских войнах, посадит править Афинами через несколько десятков лет.

Диалог «Парменид» описывает обсуждение четверкой философов следующего вопроса — един или множественен этот мир? И один и другой ответ на этот вопрос в ходе диалога неоднократно и обосновывается, и опровергается. В результате современные оценки этого произведения расходятся от того, что в нем источник всей последующей диалектики, включая Гегелевскую, до того, что это была просто выездная рекламная кампания по продвижению элейской софистики среди афинян, желающих обучать своих детей этой востребованной в демократических институтах Афин заморской премудрости.

Какое отношение имеет философский дискурс Платоновского диалога к нашей работе? Коль скоро мы собираемся описывать, моделировать или создавать сложные системы, вопрос о единственности или множественности мира становится для нас актуальным. Мы хотим включать в свою деятельность все более сложные системы и даже целые миры таких систем — растет сложность концептуальных моделей (онтологий) их предметных областей. Возникают, развиваются и усложняются специальные языки описания концептуальных моделей, есть ли предел этой сложности? Разработка больших информационных систем задыхается от сложности: сложна структура программной системы и связи между ее компонентами; сложна организация данных; поведение компонентов системы сложное: сложна его логика и сложна функциональность отдельных действий; наконец, сложно императивное программирование. Все это делает современное искусство разработки сложных систем практически недоступным для обычного человека. Осознание единства этого мира позволило бы не разрабатывать для все более сложных предметных областей, все более замысловатые языки их описаний, с последующей их нетривиальной программной реализацией, но позволило бы в кажущемся многообразии предметных областей выделять, находить, видеть их единство и применять единые методы описания, конструирования и программной реализации их математических моделей.

Что может внести информатика в решение вопроса упомянутого выше диалога Платона? На наш взгляд, этот вопрос можно отнести к утверждениям, которым посвящена теорема К. Геделя о неполноте формальных систем, включающих арифметику [2]. Мы не будем углубляться в основания математики, а вместо этого приведем блестящий на наш взгляд перевод сути теоремы Геделя на естествен-

ные языки, принадлежащий Нильсу Бору: «Есть истины двух типов: тривиальные, которые нелепо отрицать, и глубокие, для которых отрицание — тоже глубокая истина» [3].

Теорема К. Геделя и афоризм Н. Бора говорят, что мы живем в «недостроенном» мире, и всякая встреча с точкой бифуркации — Геделевским утверждением — приглашает нас к сотворчеству. Можно выбрать одну или другую ветвь бифуркации, например, столкнувшись с Пятым постулатом Евклида, выбрать евклидову или неевклидову геометрию. А можно построить синтез — метатеорию, в которой обе эти ветви находят себе место и могут быть классифицированы в рамках новой теории по сохраняемым ими при определенных преобразованиях аксиомам-инвариантам, как например разные геометрии в Эрлангенской программе Ф. Клейна [4].

Синтезом противоположных ответов на основной вопрос диалога «Парменид» можно считать фразу, любимую мистиками всех времен и народов «Одно во всем и все в Одном». Как можно применить эту фразу к обозначенной в данной работе теме изучения сложных систем? — Попробовать увидеть за многообразием предметных областей сложных систем некий единый принцип их описания, моделирования и реализации, например, универсальную структуру, способную упорядочить базисные множества (прежде всего, характеристики) сложной системы из любой предметной области или из широкого класса таких областей. Очень часто в настоящее время такой структурой является система дифференциальных уравнений, обыкновенных или в частных производных. Настолько, что многие исследователи склонны отождествлять математическое моделирование с решением уравнений математической физики на суперкомпьютерах. Об этом писал более трех десятков лет назад академик А.А. Дородницын: «Моделист находится в плену существующей математики: он пытается описать явления в новых областях с помощью известных математических структур — в основном дифференциальных уравнений…» [5].

Приведенная цитата поднимает вопрос важности для моделирования имеющегося математического языка. В то же время, всякая успешная математическая модель заметно расширяет язык дискурса в своей предметной области. Об этом много писал член-корр. РАН Ю.Н. Павловский, предложив метод гуманитарно-математического анализа сложных систем (ГМ-технология) [6]. Метод состоит в применении нескольких итераций гуманитарного анализа исследуемой предметной области, позволяющего затем построить ее математическую модель. Математическая модель обогащает язык дискурса в предметной области, что позволяет на следующей итерации проводить более глубокий гуманитарный анализ, используя расширенный моделированием язык, а затем, на основе нового гуманитар-

ного понимания предметной области, строить более адекватную математическую модель, попутно развивая и язык дискурса в этой области.

Данная работа предлагает единый способ описания предметных областей широкого класса сложных систем, реализуя в этом описании принцип "Одно во всем и все в Одном". На основании этого строится близкая методам САПР сквозная технология описания, синтеза и программной реализации моделей сложных систем — модельный синтез и модельно-ориентированное программирование [7]. Предлагается язык, отличный от дифференциальных уравнений и в некотором смысле двойственный ему. Описания реальности дифференциальными уравнениями можно считать наследием исследований, восходящих к Гераклиту и Милетским философам — искателям первоначал. Наряду с этим направлением, существовали и первые атомисты, Левкипп и Демокрит, которых можно считать основоположниками агентного моделирования, которому посвящена и данная работа.

Сложные системы. К сожалению, автору не известно определение сложной системы, с которым были бы согласны все, или хотя бы большинство исследователей, работающих в этой области. Поэтому сначала вместо определения будет приведен ряд свойств сложных систем, ключевых для данной работы. Приведем три высказывания Н.П. Бусленко из работы [8] о сложных системах.

1. 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. Компоненты сложной системы сами могут быть сложными системами.

2. В каждый момент времени элемент сложной системы находится в одном из возможных состояний; из одного состояния в другое он переходит под действием внешних и внутренних факторов.

3. Для построения синтеза поведения сложной системы необходимо дать ее компонентам возможность в полной мере проявить себя.

Данные высказывания можно считать ключом к построению излагаемой ниже теории модельного синтеза.

Первое утверждение говорит о фрактальности или самоподобии сложной системы. Но это не обязательно фрактальность папоротника М. Барнсли или треугольника В. Серпиньского. Компоненты сложной системы подобны ей самой и друг другу не обязательно внешним обликом, который у них может быть в том числе и весьма различным, а прежде всего, своей внутренней структурой, тем что все они — сложные системы.

Второе высказывание — о том, что сложные системы и их

компоненты обладают поведением. Под поведением здесь понимается способность любой компоненты совершать конечный набор действий и умение реагировать на конечный набор событий-ситуаций, задаваемых определенными сочетаниями внутренних и внешних факторов. Компонента совершает определенное свое действие в ответ на определенное событие. Заранее мы можем не знать, как поведет себя та или иная компонента, мы лишь знаем, что и в ответ на что она умеет делать. События, побуждающие компоненту к определенному действию, могут складываться из действий многих или даже всех компонент сложной системы.

Как складываются события из действий всех компонент — вопрос синтеза поведения сложной системы. Ему посвящено третье утверждение. Оно верно, но вызывает вопрос: «А как это сделать?» Ответом на него будет излагаемая ниже теория модельного синтеза.

Выше, в комментарии к первому утверждению, внешний облик системы или ее компоненты был противопоставлен ее внутренней структуре. На этом стоит остановиться подробнее. Сложную систему стоит рассматривать в трех аспектах — трех мирах. Во-первых, это физический мир — мир реализации сложной системы, ее базисные множества — то, из чего она сделана. Несомненно, это важный аспект, но он не исчерпывает всю сложную систему, как куча кирпича, по словам А. Пуанкаре, далеко не является кирпичным домом. Домом ее делает воплощение в этом кирпиче определенного архитектурного проекта. Архитектурный проект — это информационный объект, относящийся к миру форм, структур, программ, миру информатики. Он описывает как базисные множества, из которых сделан дом (в том числе и кирпичи) связаны и взаимодействуют между собой. Информационно-формальный аспект многих современных сложных систем по объему, сложности и стоимости часто сравним или даже превосходит их материальный аспект. С одной стороны, структура достаточно независима от базисных множеств — один и тот же архитектурный проект можно реализовать и в кирпиче, и в монолите, и в дереве, и во всевозможных блоках. В то же время полной независимости нет — вряд ли что-то серьезнее хижины можно построить из самана, — он просто не выдержит нагрузки. Также пересаживания мартышки, осла, козла и косолапого мишки в известной басне Крылова не делают их квартетом — для этого необходимо быть музыкантами.

Наконец, третий мир — мир идей, аксиом, инвариантов, очерчивает круг непреложных требований к системе. Например, дом должен быть пригоден для круглогодичного проживания со всеми удобствами семьи из 10 человек.

На самом деле, метод рассмотрения сложных систем в этих трех аспектах известен со времен Платона [1], но с Нового времени

обычно осуждался, как уступка идеализму. Тем не менее, даже в самое материалистическое Советское время существовали очень интересные работы, успешно применявшие этот метод, например, в социально-правовой области [9].

Для построения теории модельного синтеза, достаточно перечисленных свойств сложных систем. В дальнейшем мы дополним их, когда будем говорить об открытых (меняющих свой состав) системах, дадим и формальное определение математической модели сложной системы, как рода структуры в смысле Н. Бурбаки [10], обладающее упомянутыми свойствами и покрывающее достаточно широкий класс прикладных систем.

Гипотеза о замкнутости. Что такое гипотеза о замкнутости? По существу — это достаточно сильное предположение о том, что моделирование возможно. Мы приступаем к работе с верой в победу. Как говорил И.В. Гете: «Человек должен непреложно верить, что непостижимое на самом деле постижимо, иначе бы его исследования угасли».

У достаточно простого предположения о возможности моделирования оказывается неожиданно много важных для построения модели следствий, таких как вид агента моделирования, свойства некоторых его важных составляющих, особенности организации вычислительного процесса модели. Об этом и пойдет речь в данном разделе.

Важнейшая составляющая модели — ее характеристики, отражающие состояние моделируемой системы и состояние влияющей на систему части внешнего мира. Первые из них называются внутренними, а вторые — внешними. Гипотеза о замкнутости предполагает, что знания внутренних характеристик $\mathbf{x}(t)$ и внешних характеристик $\mathbf{a}(t)$ модели в момент t достаточно для детерминированного и однозначного вычисления ее внутренних характеристик на некотором полуинтервале времени $(t, t+\Delta t]$ положительной длины Δt. Внешние характеристики $\mathbf{a}(t)$ считаются наблюдаемыми в любой момент времени t, что тоже весьма сильное предположение.

Однозначность и детерминированность здесь относится именно к процессу вычислений. Их предметная область может быть стохастической, но организатор процесса вычислений должен определенно знать, когда включить генератор случайных чисел и какое распределение вероятности потом использовать.

Задача моделирования — составить прогноз эволюции системы на макроскопическом отрезке времени $[0, T]$. Это подразумевает вычисление значений траектории системы $\mathbf{x}(t)$ на этом отрезке. Эти вычисления могут быть выполнены только на компьютере, если система достаточно сложна, а именно о таких и идет здесь речь. Это

означает, что за конечное время мы можем вычислить значение траектории $\mathbf{x}(t)$ лишь в конечном числе точек $t_1 < t_2 < ... < t_n$, $t_i \in [0,T]$, $i = 1,...,n$. Мы должны уметь получать представление о значениях траектории в остальных точках, на основании вычисленных, например, линейно аппроксимируя промежуточные значения.

Из сказанного следует, что мы в состоянии строить лишь модели с кусочно-гладкими траекториями. На более сложные объекты не хватит компьютерного времени. Возможность конечного числа разрывов первого рода следует из того, что если система достаточно сложна, то некоторые процессы будут всегда происходить мгновенно по сравнению с продолжительностью шага моделирования, при любом разумном выборе такого шага.

Еще о разрывах: поскольку мы моделируем приближенно, шаг моделирования Δt считается столь малым, что мы различаем лишь наличие — отсутствие разрыва на отрезке шага Δt, а где конкретно он произошел, в начале, середине или конце шага Δt — не различаем. Это позволяет нам для определенности всегда относить разрыв к левому концу отрезка Δt. Из этих же соображений считаем, что на шаге моделирования Δt может быть только один разрыв (если бы их было больше — мы все равно восприняли бы их слившимися в один). Таким образом, наличие/отсутствие разрыва на шаге моделирования подобно ординарному событию в теории потоков событий.

Теперь мы можем сформулировать гипотезу о замкнутости для исследуемых сложных систем.

Назовем модель замкнутой в точке $t \in [0,T)$, если на основании значений ее внутренних $\mathbf{x}(t)$ и внешних $\mathbf{a}(t)$ характеристик

1. мы можем определить, есть ли разрыв траектории $\Delta\mathbf{x}$ в точке t и если он есть – вычислить его однозначно;

2. мы можем найти число $\Delta t > 0$, такое что на отрезке $(t, t+\Delta t] \subseteq (0,T]$, который назовем отрезком прогноза,

3. мы можем однозначно вычислить траекторию модели, начиная с точки $\{t, \mathbf{x}(t) + \Delta\mathbf{x}(t)\}$, как гладкую функцию времени на отрезке прогноза $(t, t+\Delta t]$.

Еще одним, четвертым шагом (который уже не входит в определение замкнутости модели в точке), мы можем перевести системное время вперед на Δt, и обнаружить себя в начале следующего шага моделирования, где можно снова начинать все с п. 1.

Далее, естественно предположить замкнутость модели в каждой точке $t \in [0,T)$ (иначе, если найдется точка, где модель не замкнута, как мы сможем ее продолжить за эту точку?) Такие модели будем называть локально замкнутыми на отрезке $[0,T]$.

Локальная замкнутость представляется нам необходимым условием возможности построения модели: сложно представить себе, как ее строить, если в какой-то момент времени невозможен положительный шаг вперед. Требование локальной замкнутости на отрезке $[0,T]$ покрывает каждую точку $t \in [0,T)$. связанным с ней отрезком прогноза $[t,t+\Delta t]$, на котором мы умеем вычислять траекторию системы. Это бесконечное покрытие. Сразу возникают ассоциации с компактностью. Если к локальной замкнутости добавить требование непрерывности траектории системы слева в любой точке $t \in (0,T]$, — этого будет достаточно для выделения конечного подпокрытия из бесконечного покрытия отрезка моделирования $[0,T]$ отрезками прогноза $[t,t+\Delta t]$. Без требования непрерывности траектории слева, конечного подпокрытия может не быть — могут возникать ситуации типа мухи фон Неймана [7]. Возможность выбора конечного подпокрытия из покрытия отрезка моделирования отрезками прогноза, означает теорему существования для задачи построения модели — за конечное число шагов нам удастся пройти весь отрезок моделирования $[0,T]$.

Что означает гипотеза о замкнутости для реализации системы?

То, что какова бы ни была предметная область моделирования (техническая система, конкуренция популяций, распространение эпидемии, война или социальный процесс), вся динамика модели весьма проста и одинакова для любой области. Это четырехтактный цикл, подобный циклу Карно или работе двигателя внутреннего сгорания, описанный в определении замкнутости модели в точке (с добавлением шага перевода модельного времени). Сказанное дает надежду создания универсальной программы организации имитационных вычислений модели из любой предметной области.

Такая организация вычислений похожа на работу ядра операционной системы компьютера [11], — там тоже постоянно загружен ряд системных служб, каждая из которых может выполнять конечное число заранее известных алгоритмов в соответствии с наступлением заранее известных событий — определенных сочетаний внутренних и внешних факторов.

Гипотеза о замкнутости определяет вид универсального агента моделирования, опять же вне зависимости от предметной области. Со времен Левкиппа и Демокрита и до наших дней поклонниками агентного моделирования было предложено множество самых различных типов агентов, от геометрических построений древних греков, до клеточных автоматов. И в основном все такие построения носили эвристических характер, их обоснованием была достаточность для решения задач, стоящих перед их авторами. Здесь мы хотим обратить внимание, что вид агента моделирования будет следствием принятия

гипотезы о замкнутости.

Роды структур Н. Бурбаки. В качестве основы математического языка описания сложных систем выбраны роды структур [10] в смысле Н. Бурбаки. Возникает вопрос, — почему выбран именно этот язык? Основным инструментом математического моделирования в настоящее время являются дифференциальные уравнения. Однако будучи замечательным средством моделирования физических процессов, дифференциальные уравнения не слишком хороши для описания агентных систем, обладающих поведением: корректность постановки не оставляет поведенческих альтернатив. Если все же настаивать на описании агентной системы дифференциальными уравнениями — получится кооперативная позиционная дифференциальная игра, где исходные уравнения будут ее ограничениями. Это существенно более сложный математический объект, по сравнению с системой дифференциальных уравнений, для работы с которым у нас пока нет адекватного математического аппарата (особенно, когда агентов сотни и тысячи).

Почему именно роды структур, а не похожие, но более популярные в настоящее время конструкции, как например, категории [12] или алгебраические модели [13]? Попробуем ответить на этот вопрос. Упрощенный (не имеющий вспомогательных базисных множеств) род структуры Н. Бурбаки Σ, имеет следующий синтаксис:

Σ=< базисные множества; соотношения типизации; аксиомы>.

В угловых скобках, через точку с запятой перечислены три раздела. Первый — основные базисные множества, где через запятую перечисляются эти множества. Второй — соотношения типизации, где через запятую перечисляются соотношения вида $\sigma \subset S$, где множество σ называется родовой константой, а S, так называемая ступень. Отметим, что правила построения ступеней выбраны так, чтобы соотношения типизации переносились при отображениях основных базисных множеств (морфизмах). Для этого требуется:

1. Основные и вспомогательные базисные множества считаются ступенями.

2. Если S — ступень, то $\mathfrak{B}(S)$, множество всех подмножеств S — тоже ступень.

3. Если S и S' — ступени, то и $S \times S'$ (декартово произведение S и S') — ступень.

4. Других ступеней нет.

Третий раздел рода структуры — аксиомы — произвольные истинные утверждения над базисными множествами и родовыми константами, сохраняющиеся при изоморфизмах базисных множеств [10].

Синтаксис рода структуры явно коррелирует с предложенной

выше идеей рассматривать сложные системы в трех Платоновских мирах. Основные базисные множества — это «материальный мир», в котором воплощен род структуры, но от которого он достаточно независим, поскольку базисные множества могут впоследствии подвергаться преобразованиям (морфизмам).

Соотношения типизации задают структуру на базисных множествах. Они из мира информатики и определяют, как базисные множества связаны и взаимодействуют друг с другом.

Наконец, третий раздел — аксиомы — относится к «миру идей», лежащих в основе данного рода структуры. Эти предписания и/или запреты, ограничивают построение структуры соотношениями типизации во втором разделе.

Язык родов структур также хорош для описания открытых систем, т.е. систем переменного состава, обменивающихся с окружающим миром материей, энергией и информацией: он ориентирован на сохранение соотношений типизации и аксиом, при морфизмах базисных множеств. Открытые системы могут быть описаны родами структур, базисные множества которых подвергаются морфизмам.

Можно было бы все это описывать упомянутыми выше категориями или алгебраическими моделями? На наш взгляд, — да, поскольку эти два языка описания структур можно считать языками более низкого уровня, по сравнению с Бурбаковскими родами структур. На языке более низкого уровня можно выразить все, что написано на более высокоуровневом языке, однако и усилий на это потребуется больше. Так, на языке категорий и алгебраических моделей придется вручную выбирать соотношения типизации так, чтобы они переносились при морфизмах базисных множеств и доказывать такую переносимость. Подробнее о сравнении возможностей языков родов структур и категорий можно посмотреть в работе [14].

Наконец, несколько слов о настоящем, полном определении рода структур [10], который потребуется далее. Оно рекурсивно. Для него может потребоваться уже определенная структура на вспомогательных базисных множествах (которые в дальнейшем не подлежат морфизмам). Содержательно — например, для определения рода структуры векторного линейного пространства потребуется вспомогательная структура поля действительных или комплексных чисел. В случае данной работы потребуется вспомогательное базисное множество $\mathbb{N}$ — натуральных чисел, из которого соотношение типизации $N \subset \mathbb{N}$ выделит подмножество N, которое в силу вспомогательной аксиомы $\xi : |N| = 1$, оказывается просто натуральным числом. Род структуры в этом случае обозначается $\Sigma(\mathbb{N}, \xi)$ — в

скобках указываются вспомогательные базисные множества и аксиомы — то, что в дальнейшем не подлежит морфизмам.

Род структуры M (модель). Выше было сказно, что единой точки зрения на то, что же такое сложная система, у исследователей нет. Не разрушая интригу попытками собственных определений этого термина естественного языка, дадим в этом разделе определение модели сложной системы, как формального математического объекта рода структуры M.

Скажем несколько слов о названии рода структуры и о его описании. Впервые класс таких математических объектов был описан в работе [7], как однопараметрическое семейство упрощенных (без вспомогательных базисных множеств) родов структур, и называлось оно семейством моделей-компонент. Изначально было очень сильное желание назвать эти структуры просто объектами, но данный термин уже был занят достаточно раскрученным к тому времени объектным анализом. Поэтому было выбрано название «модель-компонента», отражающее, во-первых то, что этой структурой может быть описана практически любая модель, и, во-вторых, то, что из компонент можно образовывать комплексы. На наш взгляд, название оказалось длинноватым и не слишком ярким — слабо отражающим основополагающую роль данного понятия в предлагаемой теории. В настоящее время у автора сложилось представление, что весьма непросто было бы найти в какой-либо предметной области сложную систему, которую нельзя было бы моделировать предлагаемой структурой: Л.В. Круглов в работе [15] доказал ее алгоритмическую полноту, реализовав на ней машину Тьюринга. Поэтому далее предлагается считать род структуры M (модель) формальным математическим определением понятия модели сложной системы. Тогда под сложной системой можно понимать сущность, допускающую адекватное моделирование математическим объектом рода структуры M.

Предложим здесь несколько иное, чем в [7] описание рода структуры M, — одним родом структуры, но со вспомогательным базисным множеством $\mathbb{N}$ — натуральных чисел. Род структуры в этом случае обозначается как $\Sigma(\mathbb{N},\xi)$ — в скобках указывается то, что в дальнейшем не подлежит морфизмам.

$$\Sigma(\mathbb{N},\xi) =< \mathbf{X}, M, E, \left\{M_j\right\}_{j=1}^{N}, \left\{E_j\right\}_{j=1}^{N}, \mathbb{N};$$

$$N \subset \mathbb{N},$$

$$\mathbf{x} \subset \mathbf{X},\ \mathbf{a} \subset \mathbf{X}, \tag{1}$$

$$s \subset M,\ f \subset M, \tag{2}$$

$$\left\{m_{j,real} \subset M_j \times M\right\}_{j=1}^{N}, \tag{3}$$

$$\left\{e_{j,real} \subset E_j \times E\right\}_{j=1}^{N}, \tag{4}$$

$$\left\{m_{j,in} \subset M_j \times \mathfrak{B}(\mathbf{X})\right\}_{j=1}^{N}, \tag{5}$$

$$\left\{m_{j,out} \subset M_j \times \mathfrak{B}(\mathbf{X})\right\}_{j=1}^{N}, \tag{6}$$

$$\left\{e_{j,in} \subset E_j \times \mathfrak{B}(\mathbf{X})\right\}_{j=1}^{N}, \tag{7}$$

$$\left\{sw_j \subset E_j \times M_j \times M_j\right\}_{j=1}^{N}, \tag{8}$$

$$\left\{m_j^0 \subset M_j\right\}_{j=1}^{N}, \tag{9}$$

$$\left\{p_j \subset \mathfrak{B}(M_j) \times \mathfrak{B}(E_j) \times M_j \times \mathfrak{B}(E_j \times M_j \times M_j)\right\}_{j=1}^{N}; \tag{10}$$

$$\xi : \left|N\right| = 1,$$

$$R_1 : (\mathbf{x} \cup \mathbf{a} = \mathbf{X}) \,\&\, (\mathbf{x} \cap \mathbf{a} = \varnothing),$$

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

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

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

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

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

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

$$R_8 : \left\{\left(\left(\forall e \in E_j\right)\left(\exists ! r \in M_j \times M_j\right)\left(\left\{e, r\right\} \in sw_j\right)\right) \&\right.$$

$$\left. \&\left(\left(\left\{e, r\right\} \in sw_j, \left\{\tilde{e}, \tilde{r}\right\} \in sw_j, r = \tilde{r}\right) \Rightarrow \left(e = \tilde{e}\right)\right)\right\}_{j=1}^{N},$$

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

R_{10} : аксиома однозначности вычисления характеристик сложной системы,

R_{11} : аксиома организации имитационных вычислений>.

Здесь угловая скобка < обозначает начало описания рода структуры, а парная угловая скобка > — его конец. Три раздела этого описания: базисные множества, соотношения типизации и аксиомы разделяются между собой двумя точками с запятой — после разделов базисных множеств и соотношений типизации. Обозначение

$$\{\dots_j\}_{j=1}^{N}$$

используется для краткости и означает, что содержимое скобок повторяется через запятую N раз, при этом индекс j последовательно заменяется на $1,\dots,N$. Например,

$$\{M_j\}_{j=1}^{N}$$

— есть сокращенный вариант записи $M_1,\dots,M_N$. $\mathfrak{B}(\cdot)$ — множество всех подмножеств множества, стоящего в скобках.

Анализ модели. Начнем с основных базисных множеств. $\mathbf{X}$ — множество характеристик модели. Соотношения типизации (1) и аксиома R_1 утверждают, что все $\mathbf{X}$ состоит из двух непересекающихся подмножеств: $\mathbf{x}$ — внутренних и $\mathbf{a}$ — внешних характеристик. Далее идет M — множество различных реализаций методов-элементов, элементарных умений нашей модели, которое также состоит из двух непересекающихся подмножеств: s — медленных элементов, вычисляющих гладкую эволюцию траектории на положительном отрезке прогноза и f — быстрые методы, вычисляющие величины разрывов первого рода, о чем говорят соотношения типизации (2) и аксиома R_2.

Далее $\{M_j\}_{j=1}^{N}$ — методы процессов. В отличие от M, где все методы уникальны, в M_j методы могут повторяться. Смысл повторений — например, один и тот же алгоритм может работать с разными данными. Соотношение типизации (3) и аксиома R_3 гласят, что все методы процессов берутся из единого хранилища M, у каждого там есть единственный прообраз. Точно так же

$$\{E_j\}_{j=1}^{N}$$

— события процессов. Так же события в E_j могут и повторяться, если, например, работают с разными данными. Соотношение типизации (4) и аксиома R_4 гласят, что все события процессов берутся из единого хранилища E, у каждого там есть единственный прообраз.

Соотношения типизации (5), (6) вместе с аксиомами R_5 и R_6 описывают, как методам процессов передаются характеристики, и как от них принимаются внутренние характеристики. Для контроля понимания полезно сравнить $\mathfrak{B}(\mathbf{X})$ в (6) с $\mathfrak{B}(\mathbf{x})$ в R_6 и попытаться

ответить на вопрос, почему в (6) не написано $\mathfrak{B}(\mathbf{x})$, ведь это точнее: моделью вычисляются только внутренние ее характеристики, а внешние всегда наблюдаются?

Соотношения типизации (7) вместе с аксиомой R_7 описывают, как событиям процессов передаются характеристики. Возвращают события всегда неотрицательное число, которое есть либо прогноз времени до его наступления, если положительно, либо ноль, если оно уже наступило.

Далее, $\mathbb{N}$ — вспомогательное базисное множество натуральных чисел, из которого вспомогательное соотношение типизации $N \subset \mathbb{N}$ выделяет подмножество N, которое в силу вспомогательной аксиомы $\xi : |N| = 1$, оказывается просто натуральным числом (содержательно это количество параллельных процессов в сложной системе).

Процессы определяются соотношениями типизации (10) и аксиомами R_9 (косвенно в их определения входят также соотношения (8) с аксиомами R_8, определяющие правила переключения элементов в процессах и соотношения (9), определяющие начальные элементы процессов. Каждый процесс p_j, $j = 1,...,N$, определяемый (10) и R_9, 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. Соотношения (8) и R_8 задают правила переключения элементов под влиянием событий.

Самое важное свойство сложной системы — это наличие у нее поведения, т.е. способности тем или иным заранее известным способом — методом-элементом из M, отвечать на те или иные также заранее известные воздействия внутренней и внешней среды — методы-события из E. Методы-элементы и методы-события, — это методы в программистском смысле, т.е. подпрограммы, реализующие алгоритмы действия (элементы) или реакции на определенные сочетания характеристик модели (события). И методы, и события, наряду с характеристиками, входят в базисные множества рода структуры M.

Отметим, что обычно в математических моделях базисными множествами являются только характеристики, на которых строится структура модели. Включение чего-либо кроме характеристик в базисные множества большая редкость в математическом моделировании. Тем не менее, иногда такой подход можно встретить, причем

даже в прикладных, далеких от теоретизирования работах, например, в [16].

Наличие элементарных действий M и способности реагировать на возникающие события E в базисных множествах, в дальнейшем позволит развивать геометрическую теорию поведения сложных систем. В этой связи заметим, что хотя методы-элементы M и методы-события E и являются компьютерными программами, из гипотезы о замкнутости следует их функциональность, т.е. однозначность отображений ими входящих параметров в выходящие. Стало быть, ни состояний, ни побочных эффектов эти компьютерные программы не имеют, а реализуют функции в математическом смысле. Это позволяет, во-первых, реализовывать их в функциональной парадигме программирования, которая проще нежели императивная и реализуется, и отлаживается, и даже легко параллелится; и, во-вторых, оценивать близость между собой таких методов в традиционных топологиях соответствующих функциональных пространств, что будет важно для геометрической теории поведения, например, если мы захотим преобразовывать базисные множества рода структуры М непрерывными морфизмами.

Аксиома R_{10} единственности и однозначности вычисления внутренних характеристик является следствием гипотезы о замкнутости. Однозначность отображения характеристик модели в левой точке шага моделирования во внутренние характеристики на отрезке прогноза позволяет распараллелить процесс вычисления этого отображения без конфликтов на запись (конфликт на запись — есть явное нарушение однозначности отображения). Это следствие гипотезы о замкнутости применимо к структурам М самого низкого уровня. Для сохранения аксиомы R_{10} при объединении конечного числа родов структур М в комплекс, вводятся специальные правила такого объединения [7], о которых подробнее скажем в следующем разделе.

Алгоритм имитационных вычислений. Как работает R_{11} — аксиома организации имитационных вычислений? Эта аксиома также является следствием гипотезы замкнутости. Независимо от того, какова область моделирования, динамика очень проста и одинакова для любой модели рода структуры М. Это четырехтактный цикл, подобный циклу Карно или работе двигателя внутреннего сгорания.

Сначала мы выбираем стандартный шаг моделирования Δt. На первом шаге мы знаем начальные значения внутренних характеристик и начальные методы-элементы процессов. На следующих шагах, мы имеем текущие методы-элементы всех процессов и все внутренние характеристики модели в начале шага моделирования. Внешние характеристики наблюдаемы в любой момент времени.

Далее перечислим основные этапы алгоритма.

1. Сначала мы вычисляем все события, связанные с текущими элементами всех процессов. Правила переключения (8), R_8 определяют корреляцию событий с текущими элементами процессов. Мы можем вычислять события параллельно, но для продолжения вычислительного процесса должны дождаться завершения вычисления всех событий. Если есть произошедшие события, проверяем, есть ли переходы к быстрым элементам. Если таковые имеются — запускаются быстрые элементы (они становятся текущими). Их также можно вычислять параллельно, но нужно дождаться завершения вычислений их всех, чтобы продолжить вычислительный процесс дальше. Затем мы возвращаемся к началу пункта 1. Если переходов к быстрым элементам нет — переходим к новым медленным элементам, а затем возвращаемся к началу пункта 1.

2. Если нет наступивших событий, мы выбираем ближайший $\Delta\tau$ из всех прогнозов наступления событий.

3. Если стандартный шаг моделирования Δt не превышает прогнозируемого времени до ближайшего события $\Delta t \le \Delta\tau$, мы вычисляем текущие медленные элементы со стандартным шагом Δt. В противном случае мы вычисляем их с шагом до ближайшего прогнозируемого события $\Delta\tau$. Снова можно вычислять все медленные элементы параллельно, ожидая завершения последнего.

4. Возвращаемся к пункту 1.

На основании приведенных правил, можно написать универсальную компьютерную программу, которая организовывала бы имитационные вычисления для любой модели рода структуры М.

Комплексы моделей сложных систем и модельный синтез. Зачем нужно приведенное выше формальное описание модели сложной системы? Например, чтобы на языке теории множеств доказать, что результат объединения конечного числа моделей сложных систем в комплекс, сам является моделью сложной системы (т.е. имеет определенный выше род структуры М), а также уточнить в ходе этого доказательства правила такого объединения. Такое доказательство приведено в работе [7]. Здесь воспроизведем его наиболее важную часть — правила объединения сложных систем в комплекс. Факт замкнутости рода структуры М, относительно объединения нескольких его представителей в комплекс, вместе с аксиомой R_{11} единого способа организации вычислительного процесса для любого объекта рода структуры М и создают теорию Модельного синтеза. Название с одной стороны, вызывает несомненные ассоциации с Объектным анализом, а с другой стороны, — и определенное ему противопоставление (на самом деле имеет место двойственность).

Итак, пусть мы имеем конечное число Q объектов рода структуры M. Как мы будем объединять их в комплекс? Здесь на самом деле проблему представляет лишь объединение множеств их характеристик

$$\mathbf{X}_i, i = 1,...,Q.$$

Для понимания правил объединения, сначала предположим, что

$$\mathbf{X}_i \cap \mathbf{X}_j = \varnothing, \forall i, j : i \neq j; 1 \leq i, j \leq Q.$$

В этом случае правила объединения в комплекс весьма просты: базисные множества

$$\mathbf{X}_i, M_i, E_i, 1 \leq i \leq Q$$

объединяются по правилам объединения множеств:

$$\mathbf{X}' = \bigcup_{i=1}^{Q} \mathbf{X}_i, \quad M' = \bigcup_{i=1}^{Q} M_i, \quad E' = \bigcup_{i=1}^{Q} E_i.$$

Вспомогательные родовые константы складываются:

$$N' = \sum_{i=1}^{Q} N_i.$$

Очевидно,

$$N' \subset \mathbb{N} \text{ и } \left|N'\right| = 1.$$

Все, что в определении рода структуры M находится в фигурных скобках теперь имеет верхний индекс закрывающей скобки N': — $\{\bullet\}_{j=1}^{N'}$. Последнее означает, что мы объединяем в новом объекте все процессы исходных объектов и все относящиеся к этим объектам соотношения типизации. Из-за того, что мы объединяли базисные множества методов и событий

$$M_i, E_i, 1 \leq i \leq Q$$

и не предполагали, что они попарно не пересекаются, возможно, некоторые из соотношений типизации (3), (4) и аксиом R_3, R_4 придется перекоммутировать на другие реализации этих методов или событий. Например, если хранилища методов и событий

$$M_i, E_i, 1 \leq i \leq Q$$

были распределены в сети с разными IP-адресами. Тем не менее, вид этих соотношений и аксиом после перекоммутации не меняется.

Вот, собственно и все — выполнение всех аксиом и соотношений типизации на новых базисных множествах проверяется легко, хотя и достаточно громоздко [7].

Тем не менее, полученный в предположении $\mathbf{X}_i \cap \mathbf{X}_j = \varnothing$ комплекс не слишком интересен содержательно — в нем отсутствует системный эффект — все модели, хотя они и объединены в комплекс, будут работать точно так же, как если бы они работали сами по себе. А самое интересное в сложной системе — это именно системный эффект.

Системный эффект дают именно пересечения множеств характеристик компонент комплекса. Поэтому придется рассматривать такие пересечения, они бывают трех типов:

1. $\mathbf{x}_i \cap \mathbf{a}_j \neq \varnothing, i \neq j$.

Этот способ пересечения характеристик можно считать штатным, посредством которого обычно устанавливается связи между компонентами комплекса при его проектировании. Он означает, что некоторые внешние характеристики одних компонент комплекса явно вычисляются другими его компонентами, как их внутренние характеристики. При этом, конечно, предполагается, что моделируются эти внутренние характеристики адекватно. В этом случае вычисляемые внешние характеристики компонент исключаются из внешних характеристик комплекса, а в соотношениях типизации (5), (7) и аксиомах R_5, R_7 эти исключенные внешние характеристики заменяются вычисляющими их внутренними. Такая перекоммутация характеристик не нарушает формы соотношений (5), (7) и аксиом R_5, R_7.

2. $\mathbf{a}_i \cap \mathbf{a}_j \neq \varnothing, i \neq j$.

Это означает, что у некоторых компонент комплекса имеются одинаковые внешние характеристики. Исходя из того, что внешние характеристики описывают влияния внешнего мира на наши модели и из единства внешнего мира, считаем, что пересекаются тождественные характеристики. Поэтому множество внешних характеристик комплекса можно получить из множеств внешних характеристик его компонент обычным объединением множеств:

$$\mathbf{a}' = \bigcup_{i=1}^{Q} \mathbf{a}_i .$$

При этом придется перекоммутировать часть соотношений типизации (5), (7) и аксиом R_5, R_7, заменяя некоторые внешние переменные тождественными, что не меняет форму соотношений (5), (7) и аксиом R_5, R_7.

3. Наконец, самый сложный и неприятный случай:

$$\mathbf{x}_i \cap \mathbf{x}_j \neq \varnothing, i \neq j .$$

Он означает, что пересекаются внутренние характеристики компонент. Такое может случиться, например, если мы объединяем в комплекс десяток компонент, каждая из которых вычисляет прогноз погоды или курса валют на завтра. Неприятен он тем, что явно нарушает аксиому R_{10} единственности и однозначности вычисления внутренних характеристик. На самом нижнем уровне «атомов» ее выполнение гарантировала гипотеза о замкнутости. На аксиоме R_{10} основана параллельность вычислительного процесса рода структуры М, поэтому очень хотелось бы сохранить ее и в комплексе. Решить данную проблему можно следующим образом. Пусть внутренние переменные S компонент $\mathbf{x}_{j_1},...,\mathbf{x}_{j_S}$ вычисляют одну и ту же внутреннюю характеристику комплекса $\mathbf{x}'$. Введем новую компоненту с S внешними переменными $\tilde{\mathbf{a}}_1,...,\tilde{\mathbf{a}}_S$ и одной внутренней $\tilde{\mathbf{x}}$. Содержание новой компоненты – в вычислении $\tilde{\mathbf{x}}$ на основании $\tilde{\mathbf{a}}_1,...,\tilde{\mathbf{a}}_S$ одним из множества возможных способов: свертка, случайный выбор, аналитика на основании некой математической модели и т.д. В комплексе коммутируются в смысле п. 1 характеристики $\mathbf{x}_{j_1},...,\mathbf{x}_{j_S}$ с $\tilde{\mathbf{a}}_1,...,\tilde{\mathbf{a}}_S$, а внутренняя характеристика $\tilde{\mathbf{x}}$ становится внутренней характеристикой $\mathbf{x}'$ комплекса. Это снимает неоднозначность и угрозу аксиоме R_{10}. Остается еще замена $\mathbf{x}_{j_1},...,\mathbf{x}_{j_S}$ на $\mathbf{x}'$ в соотношениях типизации (5), (7) и аксиомах R_5, R_7. Такая замена не меняет формы этих соотношений и аксиом.

Описания действий при наличии пересечений характеристик $\mathbf{X}_i \cap \mathbf{X}_j = \varnothing$ и завершают правила объединения нескольких объектов рода структуры М в комплекс. Примеры проверки выполнения для комплекса всех аксиом и соотношений типизации, как уже говорилось выше, приводятся в работе [7].

Как работает Модельный синтез?

Ему должен предшествовать анализ, основанный на свойстве самоподобия сложной системы. Выделяются компоненты сложной системы, затем компоненты их компонент и т.д. На каком-то этапе следует остановиться и назначить компоненты этого самого низкого уровня агентами — нашими атомами. Как правило, это происходит не из-за невозможности дальнейшего выделения компонент, а из-за достаточности уже выделенных, для желаемого уровня подробности моделирования.

Может возникнуть вопрос о формализации выделения компонент

сложной системы. Здесь нам известно как минимум три подхода. В геометрической теории декомпозиции Ю.Н. Павловского [17] рассмотрены два двойственных между собою способа декомпозиции, — на подобъекты и на фактор-объекты. Однако здесь в обоих случаях имеется в виду точная декомпозиция, где декомпозированные объекты не связаны между собой. Такая декомпозиция встречается в жизни достаточно редко, а когда встречается — свидетельствует об определенном вырождении исследуемой системы. В любимой многими инженерами работе Г. Крона [18] рассматривается приближенная декомпозиция, где имеют место связи между компонентами типа вход — выход, что не нарушает системного эффекта. Однако ряду математиков ее изложение представляется недостаточно формальным. Работа [19] рассматривает вопрос выделения компоненты в математическом объекте рода структуры М. Ее вывод — в качестве компоненты можно взять любое количество процессов (9), (10), R_9 исходной структуры начиная от одного, меньшее полного количества ее процессов, при некоторых, не слишком обременительных, дополнительных условиях. Этот вывод неплохо согласуется с предлагаемыми в [18] методами выделения частей сложной системы. На «месте разреза» и у новой, и у оставшейся компонент возникают внешние характеристики, которые ранее вычислялись как внутренние другой компонентой. Удачность или неудачность такого выделения, по-видимому, можно оценить как раз количеством этих новых внешних характеристик, чем их меньше — тем удачнее будет такая приближенная декомпозиция. Это довольно хорошо согласуется с интуитивным представлением о компоненте сложной системы, как о сущности, гораздо сильнее связанной внутри себя, нежели с чем-то внешним, например, с другими компонентами.

На основании знаний устройства и поведения найденных агентов, описываем их родом структуры М. С учетом проделанного ранее анализа компонент и знания связей между агентами, они объединяются в комплексы. Замкнутость семейства математических объектов рода структуры М относительно объединений в комплексы, позволяет пройти весь обратный путь наверх, — объединяя компоненты, комплексы которых тоже будут объектами рода структуры М. При этом возможно построение весьма сложных конструкций, но вопрос как организовать их вычисления не возникает, — имитационные вычисления любого объекта рода структуры М организовывает одна и та же универсальная программа.

Модельный синтез: выводы. Род структуры *M*, как основа языка дискурса в области сложных систем. Теория модельного синтеза позволяет, во-первых, описать сложную систему, начиная от

описания родами структуры М агентов атомарного уровня, затем последовательно описывая по установленным правилам их комплексы, которые сами оказываются объектами рода структуры М, до самого верхнего уровня всей системы, также являющейся объектом рода структуры М. Во-вторых, полученное описание сложной системы является ключом к ее автоматическому синтезу — по описанию комплекса, с указанием его состава и коммутации компонент, можно автоматически создать его описание как объекта рода структуры М, по указанным выше правилам. При этом, самыми частыми действиями занимающейся этим компьютерной программы, будут перекоммутации некоторых соотношений типизации родов структуры М — действия одного и того же типа, хорошо поддающиеся программированию. В-третьих, автоматический синтез сложной системы позволяет предложить новую парадигму программирования для программной реализации модели сложной системы — модельно-ориентированное программирование, где универсальной единицей построения программной системы будет объект рода структуры М. Л.В. Кругловым в работе [15] показано, что класс программ поведения, задаваемый соотношениями типизации (8)–(10) и аксиомами R_8, R_9 является алгоритмически полным, что обосновывает алгоритмическую полноту предлагаемого метода. Парадигму модельно-ориентированного программирования предполагается осветить в продолжении данной статьи в ближайших номерах журнала.

Еще одним интересным следствием модельного синтеза, которое также предполагается раскрыть в продолжении этой статьи, является то, что он дает новый язык дискурса в области сложных, в том числе и открытых (меняющих свой состав) систем. Такие системы часто являются объектами изучения описательных, например, социальных наук. Для дискурса в области сложных открытых систем предлагается язык рода структуры М. Как это может работать? Приведем следующую аналогию.

Н. Бурбаки в работе [10] предложены языки трех уровней для описания всевозможных математических объектов (еще до появления программирования, алгоритмических языков и трансляторов).

Это язык самого низкого уровня с исходным алфавитом из семи специальных символов:

$$\square,\ \tau,\ \vee,\ \neg,\ =,\ \in,\ \ominus,$$

надстрочного знака связи символа τ (тот, тот самый) с символом $\square$ (который), плюс неконкретизируемый в [10] естественный алфавит. Настоящее Бурбаковское описание математического объекта существует именно на этом языке. На нем необыкновенно просты

доказательства, но для обычного человеческого ума он совершенно непригоден — в нем «слишком много буков». Описанное на нем сколько-нибудь содержательное понятие невозможно ни удержать в уме, ни, тем более, оперировать с ним. Ситуация похожа (с очень большим усилением) на двоичную арифметику. Это сверхпростая арифметика, однако обилие знаков делает ее пригодной только для компьютера, но не для человека.

На промежуточном уровне предлагается язык сокращений, где привычные математические символы (например, кванторы $\exists$, $\forall$, пустое множество $\varnothing$, числа и многие другие обычные математические конструкции) описываются на языке низкого уровня.

Наконец, язык высокого уровня. На нем пишутся почти обычные по виду математические выражения, формулируются определения и доказываются теоремы. Их вполне можно и воспринимать в обычном смысле, по крайней мере, существует ряд работ (например, эта, или [7]), где Бурбаковские конструкции, например, роды структур, рассматриваются на основе традиционной, а не Бурбаковской аксиоматики теории множеств. Однако все Бурбаковские конструкции, наряду с обычным общематематическим восприятием, допускают и специфическую Бурбаковскую интерпретацию — их можно «разбурбачить» — транслировать в язык низкого уровня, где все они приобретают совершенно формальный смысл, становятся малопригодными для восприятия человеком из-за слишком большого объема записи, но зато дают возможность проверки конструкций языка высокого уровня.

Выводы. Предлагаемая теория модельного синтеза позволяет формально описать практически любую агентную систему, в виде объекта описанного выше рода структуры М. В том числе и социальные системы, традиционно являющиеся предметом изучения описательных наук, например такие как, страна, экономика, предприятие, театр военных действий, политическая партия, социальный слой, множество читателей этой статьи. Упомянутая выше алгоритмическая полнота объектов рода структуры М позволяет заключить, что вряд ли найдется сложная система, модель которой можно построить на компьютере, но нельзя представить объектом этого рода структуры.

Род структуры М — это язык низкого уровня описаний таких систем. Далее, можно вводить язык промежуточного уровня. Например, под поведением сложной системы понимать набор ее умений M, (3), R_3; событий E, (4), R_4 — ситуаций на которые система умеет реагировать; программу ее поведения (8)–(10), R_8, R_9, т.е. выбора действий, в зависимости от происходящих событий.

Стало быть, гуманитарное по виду изложение, которое будет

предложено в продолжении данной работы есть не просто гуманитарный анализ проблем в смысле работы [6] (хотя воспринимать его таким образом отнюдь не возбраняется), — его вполне можно трактовать как сокращенный математический текст на языке высокого уровня, который в любой момент можно транслировать в язык низкого уровня — формальное описание соответствующей модели в виде рода структуры M в смысле Н. Бурбаки, уточняя этим гуманитарное высказывание.

ЛИТЕРАТУРА

[1] Платон. *Диалоги*. Москва, Мысль, 1986, 607 с.

[2] Ершов Ю.Л., Палютин Е.А. *Математическая логика*. Москва, Наука, 1987, 336 с.

[3] Бор Н. *Атомная физика и человеческое познание*. Москва, Издательство иностранной литературы, 1961, 151 с.

[4] Норден А.П. *Об основаниях геометрии: сборник классич. работ по геометрии Лобачевского и развитию ее идей*. Москва, Гостехиздат, 1956, 527 с.

[5] Дородницын А.А. Информатика: предмет и задачи. *Вестник АН СССР*, 1985, № 2, с. 85–89.

[6] Малинецкий Г.Г., Коротаев А.В. *Проблемы математической истории: историческая реконструкция, прогнозирование, методология*. Москва, URSS, 2008, с. 246.

[7] Бродский Ю.И. *Модельный синтез и модельно-ориентированное программирование*. Москва, ВЦ РАН, 2013, 142 с.

[8] Бусленко Н.П. *Моделирование сложных систем.* Москва, Наука, 1978, 400 с.

[9] Леванский В.А. *Моделирование в социально-правовых вопросах*. Москва, Наука, 1986, 158 с.

[10] Бурбаки Н. *Теория множеств.* Москва, Мир, 1965, 456 с.

[11] Карпов В.Е., Коньков К.А. *Основы операционных систем: курс лекций*. Москва, Физматкнига, 2019, 328 с.

[12] Маклейн С. *Категории для работающего математика*. Москва, Физматлит, 2004, 352 с.

[13] Бускаран Э. *Теория моделей и алгебраическая геометрия*. Москва, МЦНМО, 2008, 280 с.

[14] Данилов Н.Ю. О взаимосвязи декомпозиционных свойств исчисления родов структур и теории категорий. *Моделирование, декомпозиция и оптимизация сложных динамических процессов*, 1996, т. 11, № 1–2 (11), с. 49–62.

[15] Kruglov L.V., Brodsky Yu.I. Model-oriented programming. *Proceedings of CBU in Natural Sciences and ICT*, 2021, vol. 2, pp. 63–67. DOI: 10.12955/pns.v2.154

[16] Cohn A., Maréchal M.A., Tannenbaum D., Zünd C.L. Civic honesty around the globe. *Science*, vol. 365, iss. 6448, pp. 70–73.

[17] Павловский Ю.Н., Смирнова Т.Г. *Введение в геометрическую теорию декомпозиции*. Москва, Фазис, 2006, 169 с.

[18] Крон Г. *Исследование сложных систем по частям — диакоптика*. Москва, Наука, 1972, 542 с.

[19] Бродский Ю.И. О приближенной декомпозиции модели-компоненты. *Моделирование, декомпозиция и оптимизация сложных динамических процессов*, 2014, т. 29, № 1 (29), с. 119–127.

[20] Димитриенко Ю.И., Димитриенко О. Ю. Модель многомерной деформируемой сплошной среды для прогнозирования динамики больших массивов индивидуальных данных. *Математическое моделирование и численные методы,* 2016, № 1, с. 105–122.

Статья поступила в редакцию 12.06.2022

Ссылку на эту статью просим оформлять следующим образом:
Бродский Ю.И. Структурная теория сложных систем. Модельный синтез. *Математическое моделирование и численные методы*, 2022, № 3, с. 98–123.

Бродский Юрий Игоревич — д-р физ.-мат. наук, ведущий научный сотрудник ФИЦ ИУ РАН. e-mail: yury_brodsky@mail.ru

Structural theory of complex systems. Model synthesis

© Yu.I. Brodsky

FRC CSC RAS, Moscow, 119333, Russia

The purpose of this work is to organize from a unified viewpoint the results of the author's work in the field of the structural theory of complex systems modeling and the practice of their implementing of the last two decades. We propose a formal definition of the complex system computer model, as a species of structure in the sense of N. Bourbaki — the M (System) species of structure, based on the humanitarian analysis of the complex systems key properties, recognized by a number of authoritative researchers and practicians in this field, and the assumption of the possibility of constructing a mathematical computer model of a complex system, — the closure hypothesis. The class of mathematical objects defined by the M species of structure has the following two properties: a complex created by combining a finite number of mathematical objects of the M species of structure, according to the certain rules, is itself a mathematical object of the same M species of structure. The computation organization process is same for all the mathematical objects of the M species of structure and therefore can be implemented by a single universal program for the simulation calculations organization. The presence of these two properties of the M species of structure representatives allows us to build an end-to-end technology for the description, synthesis and software implementation of the complex systems models — Model Synthesis and Model-Oriented Programming. By studying the morphisms of the M species of structure base sets of the model constructed with the model synthesis help, and the invariants limiting such morphisms, we obtain a formal mathematical language for the study of complex open (changing their composition) systems. By conducting a traditional humanitarian discourse, one can always correlate it with the corresponding object of the M species of structure — translating the higher-level language of humanitarian concepts into mathematical language. The proposed theory has a practical application in the field of development, description and implementation of complex software systems. A new programming paradigm is proposed — Model-Oriented Programming, which is a complete implementation of CAD methods in programming. When developing a software system, it is possible to stay within the framework of declarative programming, avoiding imperative, which greatly simplifies both its development and implementation, and subsequent debugging.

Keywords: *complex systems, structural theory, model synthesis, geometric theory of behavior, model-oriented programming, formalization of social sciences*

REFERENCES

[1] Plato. *Dialogi* [Dialogues]. Moscow, Mysl Publ., 1986, 607 p.

[2] Ershov Yu.L., Palyutin E.A. *Matematicheskaya logika* [Mathematical logic]. Moscow, Nauka Publ., 1987, 336 p.

[3] Bohr N. *Atomic physics and human knowledge*. John Wiley & Sons Publ., 1958, 101 p.

[4] Norden A.P. *Ob osnovaniyah geometrii: sbornik klassicheskih rabot po geometrii Lobachevskogo i razvitiyu ee idej* [On the foundations of geometry: a collection of classical works on Lobachevsky's geometry and the development of its ideas]. Moscow, Gostekhizdat Publ., 1956, 527 p.

[5] Dorodnicyn A.A. Informatika: predmet i zadachi [Informatics: subject and tasks]. *Vestnik AN SSSR* [Bulletin of the USSR Academy of Sciences], 1985, № 2, pp. 85–89.

[6] Malinetsky G.G., Korotaev A.V. *Problemy matematicheskoj istorii: istoricheskaya rekonstrukciya, prognozirovanie, metodologiya* [Problems of mathematical history: historical reconstruction, forecasting, methodology]. Moscow, URSS, 2008, p. 246.

[7] Brodsky Yu.I. *Model'nyj sintez i model'no-orientirovannoe programmirovanie* [Model synthesis and model-oriented programming]. Moscow, CC of RAS Publ., 2013, 142 p.

[8] Buslenko N.P. *Modelirovanie slozhnyh sistem* [Modeling of complex systems]. Moscow, Nauka Publ., 1978, 400 p.

[9] Levansky V.A. *Modelirovanie v social'no-pravovyh voprosah* [Modeling in social and legal issues]. Moscow, Nauka Publ., 1986, 158 p.

[10] Burbaki N. *Teoriya mnozhestv* [Set theory]. Moscow, Mir Publ., 1965, 456 p.

[11] Karpov V.E., Konkov K.A. *Osnovy operacionnyh sistem: kurs lekcij* [Fundamentals of operating systems: a course of lectures]. Moscow, Fizmatkniga Publ., 2019, 328 p.

[12] Mac Lane S. *Categories for the working mathematician*. New York, Springer, 1978, 317 p.

[13] Buskaran E. *Teoriya modelej i algebraicheskaya geometriya* [Model theory and algebraic geometry]. Moscow, MCNMO Publ., 2008, 280 p.

[14] Danilov N.Yu. O vzaimosvyazi dekompozicionnyh svojstv ischisleniya rodov struktur i teorii kategorij [On the relationship of the decomposition properties of the calculus of genera of structures and category theory]. *Modelirovanie, dekompoziciya i optimizaciya slozhnyh dinamicheskih processov* [Modeling, decomposition and optimization of complex dynamic processes], 1996, vol. 11, no. 1–2 (11), pp. 49–62.

[15] Kruglov L.V., Brodsky Yu.I. Model-oriented programming. *Proceedings of CBU in Natural Sciences and ICT*, 2021, vol. 2, pp. 63–67. DOI: 10.12955/pns.v2.154

[16] Cohn A., Maréchal M.A., Tannenbaum D., Zünd C.L. Civic honesty around the globe. *Science*, vol. 365, iss. 6448, pp. 70–73.

[17] Pavlovsky Yu.N., Smirnova T.G. *Vvedenie v geometricheskuyu teoriyu dekompozicii* [Introduction to the geometric theory of decomposition]. Moscow, Phasis Publ., 2006, 169 p.

[18] Kron G. *Issledovanie slozhnyh sistem po chastyam — diakoptika* [Investigation of complex systems in parts — diacoptics]. Moscow, Nauka Publ., 1972, 542 p.

[19] Brodsky Yu.I. O priblizhennoj dekompozicii modeli-komponenty [On the a pproximate decomposition of the model-components]. *Modelirovanie, dekompoziciya i optimizaciya slozhnyh dinamicheskih processov* [Modeling, decomposition and optimization of complex dynamic processes], 2014, vol. 29, no. 1 (29), pp. 119–127.

[20] Dimitrienko Y.I., Dimitrienko O.Y. A model of multidimensional deformable continuum for forecasting the dynamics of large scale array of individual data. *Mathematical Modeling and Computational Methods*, 2016, no. 1, pp. 105–122.

Brodsky Yu.I., Dr. Sci. (Phys. — Math.), leading researcher of the FRC CSC RAS. e-mail: yury_brodsky@mail.ru