DOI: 10.25728/mlsd.2022.0428

СТРУКТУРНАЯ ТЕОРИЯ, КАК ЯЗЫК ИССЛЕДОВАНИЯ ОТКРЫТЫХ КРУПНОМАСШТАБНЫХ СИСТЕМ

Бродский Ю.И.
Федеральный исследовательский центр «Информатика и управление» РАН, Россия, г. Москва, ул. Вавилова, д. 44/2,
yury_brodsky@mail.ru

Аннотация: Структурная теория позволяет увидеть одну и ту же универсальную конструкцию модели в любых предметных областях моделирования. Применение геометрической теории является естественным продолжением структурного метода: рассматриваются морфизмы базисных множеств и ищутся инварианты, позволяющие сложной системе оставаться самой собой при таких морфизмах.

Ключевые слова: сложные системы, структурная теория, модельный синтез, геометрическая теория поведения, формализация описательных наук.

Введение

Диалог Платона «Парменид» [1] описывает обсуждение четверкой философов следующего вопроса – един или множествен 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 диалектики, включая Гегелевскую, до того, что это была просто выездная рекламная кампания по продвижению элейской софистики среди афинян, желающих обучать своих детей этой заморской премудрости, востребованной в демократических институтах Афин.

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? 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 естественные 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 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 гуманитарного 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.

1 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 объединенные 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.

2 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 [11] или алгебраические модели [12]. 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 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 (автоматического их переноса там может и не быть). Подробнее о сравнении возможностей языков родов структур и категорий можно посмотреть в работе [13].

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.

3 Род структуры CS (сложная система)

Скажем несколько слов о названии структуры и о ее 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 описать предлагаемой структурой: Л.В. Круглов в работе [14] доказал ее алгоритмическую полноту, реализовав на ней машину Тьюринга. Поэтому далее предлагается считать род структуры CS (Complex System) формальным математическим определением понятия модели сложной системы. Тем не менее, в дальнейшем тексте под словами «сложная система» будем понимать термин естественного языка, а под «родом структуры CS» – формально определенный класс математических объектов.

Предложим здесь несколько иное чем в [7] описание рода структуры CS, – 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.

$$
\begin{gathered}
\Sigma(\mathbb{N}, \xi) =< \mathbf{X}, M, E, \{M_j\}_{j=1}^{N}, \{E_j\}_{j=1}^{N}, \mathbb{N}; \\
N \subset \mathbb{N}, \\
\mathbf{x} \subset \mathbf{X}, \mathbf{a} \subset \mathbf{X}, \qquad (1)
\end{gathered}
$$

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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 $\left\{ \ldots_j \right\}_{j=1}^{N}$ используется для краткости и означает, что содержимое скобок повторяется через запятую N раз, при этом индекс j последовательно заменяется на $1, \ldots, N$. Например, $\left\{ M_j \right\}_{j=1}^{N}$ – есть сокращенный вариант записи $M_1, \ldots, M_N$. $\mathfrak{B}(\cdot)$ – множество всех подмножеств множества, стоящего в скобках.

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 $\left\{M_j\right\}_{j=1}^N$ – 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 $\left\{E_j\right\}_{j=1}^N$ – 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, $j = 1, \dots, N$, 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 CS.

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, такой подход можно встретить, причем даже в прикладных, далеких от теоретизирования работах, например, в [15].

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 CS 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 CS самого низкого уровня. Для сохранения аксиомы R_{10} при объединении конечного числа родов структур CS в комплекс, вводятся специальные правила такого объединения [7], о которых подробнее скажем в следующем разделе.

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 CS. 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

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 CS.

4 Модельный синтез

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 [16] 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 [17] 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 [18] рассматривает вопрос выделения компоненты в математическом объекте рода структуры CS. Ее вывод – в качестве компоненты можно взять любое количество процессов (9), (10), R_9 исходной структуры начиная от одного, меньшее полного количества ее процессов. 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 CS. С учетом проделанного ранее анализа компонент и знания связей между агентами, они объединяются в комплексы. Замкнутость семейства математических объектов рода структуры CS относительно объединений в комплексы, позволяет пройти весь обратный путь наверх, – объединяя компоненты, комплексы которых тоже будут объектами рода структуры CS. При этом возможно построение весьма сложных конструкций, но вопрос как организовать их вычисления не возникает, – имитационные вычисления любого объекта рода структуры CS организовывает одна и та же универсальная программа.

5 Структурная теория, как язык дискурса в области сложных систем

Может возникнуть вопрос, зачем в предыдущих разделах приведено множество формул, если в последующих дискурс ведется на достаточно «гуманитарном» уровне? Для прояснения этого факта приведем следующую аналогию.

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

Это язык самого низкого уровня с исходным алфавитом из семи специальных символов: $\square$, τ, $\vee$, $\neg$, $=$, $\in$, $\ominus$, плюс надстрочный знак связи символа τ (тот) 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 букв», для возможности осознать описанное на нем сколько-нибудь содержательное понятие. Ситуация очень похожа на двоичную арифметику. Эта арифметика сверхпроста, зато обилие знаков делает ее пригодной лишь для компьютера, но не 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 Бурбаковский смысл. Правда, как правило, становятся от этого совершенно недоступными для обычного восприятия человеком, из-за слишком большого объема записи.

Предлагаемая теория модельного синтеза позволяет формально описать практически любую агентную систему, в виде объекта описанного выше рода структуры CS. 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.

Род структуры CS – это язык низкого уровня таких описаний. 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.

Л.В. Кругловым в работе [14] показано, что класс программ поведения, задаваемый соотношениями типизации (8) – (10) и аксиомами R_8, R_9 является алгоритмически полным. Поэтому, например, на языке высокого уровня можно говорить о «сохранении законов» в сложной открытой системе, как о достаточно сложном алгоритме управления системой, сохраняющем ее структуру и систему аксиом-инвариантов в определенных пределах, и тем не менее, формально описываемом соотношениями (8) – (10) и аксиомами R_8, R_9.

Стало быть, гуманитарный по виду дискурс последующих разделов данной работы есть не просто гуманитарный анализ проблемы 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 сложной системы в виде рода структуры CS в смысле Н. Бурбаки.

6 Геометрическая теория поведения сложных открытых систем

Изучая свойства сложных систем, обратим внимание на такой важный их класс, как открытые (т.е. меняющие свой состав) сложные системы. Такие системы изучались школой И. Пригожина [19], где считалось, что суть их самоорганизации в диссипативности и термодинамической неравновесности. В нашей стране открытые системы изучались школой С.П. Курдюмова [20], считавшей, что суть – в нелинейности и возможности выходов на режимы с обострениями. Ю.Н. Павловский во многих своих работах, посвященных моделированию боевых действий (напр., [21]) отмечал, что сложные системы в условиях вооруженной борьбы прежде всего стремятся сохранять свою структуру. Следуя за ним (и не отрицая при этом значимости фактов нелинейности, диссипативности и неравновесности), автор считает, что самоорганизация сложных открытых систем – их modus vivendi – пока открытая сложная система существует, она стремится сохранять законы своего существования и доколе ей это удается – она существует. В противном случае, ее ожидает переход в иное качество, скорее всего это распад на составляющие с последующим их поглощением более успешными системами. Понять это помогает геометрическая теория поведения.

Вокруг мы встречаем много открытых систем. Например, таковыми являемся мы сами – наш атомарный состав в течение жизни несколько раз меняется практически полностью. Состав призывников в армии меняется наполовину раз в полгода, а если армия участвует в боевых действиях – ее состав может существенно обновиться на временах порядка месяца и даже недели. В университете за год бакалавриат обновляется на четверть, а магистратура наполовину. Определенная текучесть кадров имеет место на любом предприятии. В стране раз в 25 – 30 лет меняются поколения, определяющие ее облик. Тем не менее, обычно мы склонны считать подобные сложные открытые системы «теми же самыми», почему? Что в них остается, что заставляет считать их теми же самыми?

Можно ответить, что той же самой остается (или меняется на порядки медленнее) их структура. Однако, с точки зрения теории Модельного синтеза, такой ответ будет не слишком содержательным, так как с ее позиций практически все в этом мире – объекты рода структуры CS и неудивительно поэтому, что их структура не меняется. Попробуем средствами гуманитарного анализа [6] уточнить, какая часть структуры позволяет воспринимать объект тем же самым, несмотря на его переменный состав. При взаимодействии с организациями чаще всего нам все равно, кто там работает, важно – какие услуги или товары и по каким ценам, может нам предоставить эта организация. Тем, кто поступает в университет, все равно, кто там учится, да и то кто учит не слишком существенно, если известен общий высокий уровень преподавателей и студентов. Важно – чему и насколько качественно там учат, как организован процесс обучения и проживания. Точно так же, гражданам неважно, кто сейчас служит в рядах армии, важно может или нет армия защитить страну. Тому, кто совершает покупку в гипермаркете, безразлично, кто конкретно сидит на кассе, – важно, чтобы касс было достаточно, и работали они быстро и четко. Средний пользователь персоналки вполне может не знать, какой у этого компьютера процессор, AMD, Intel или VIA. А вот разницу между операционными системами Windows, Mac OS и Linux (и даже их разными версиями), которые можно на него загрузить, заметит обязательно: и интерфейс меняется, и любимые приложения несколько иные (если вообще находятся), и результат их работы отличается.

Этот краткий гуманитарный анализ оставляет впечатление, что мы считаем сложную открытую систему той же самой, если не меняется (или меняется на порядки медленнее состава) ее поведение, которое при взаимодействии с системой для нас важнее, чем реализующий это поведение состав. При этом под поведением мы можем разуметь не только понятие естественного языка, имеющее согласно В.В. Налимову [22] континуальное поле смыслов, но и формальное математическое определение, задаваемое прежде всего, базисными множествами M (возможные действия-реакции системы) и E (возможные события, на которые система должна реагировать), соотношениями типизации (8) – (10) и аксиомами R_8, R_9 (связь событий с вызываемыми ими действиями и процессы – программы поведения), но косвенно также включающее соотношения типизации (3) – (7) и аксиомы R_3 – R_7 (реализация действий и событий, их связь с характеристиками системы). Поведение имеет любой объект рода структуры CS, поэтому применительно к сложной системе можно говорить о разных уровнях ее поведения: индивидуальное поведение агентов – «атомов» самого низкого уровня, коллективное поведение всевозможных промежуточных комплексов, наконец, коллективное поведение системы в целом, т.е. комплекса самого высокого из рассматриваемых уровня.

Как можно моделировать изменения состава и поведения в объектах рода структуры CS? Для объектов самого нижнего уровня «атомов» – это прежде всего морфизмы множеств M и E, хотя и характеристики $\mathbf{X}$ также могут подвергаться преобразованиям. Характеристики модели – числа, методы-элементы и методы-события – компьютерные программы, но в силу гипотезы о замкнутости – однозначные, без побочных эффектов и состояний, т.е. эквивалентные математическим функциям, поэтому близость между ними можно оценивать с помощью топологий соответствующих функциональных пространств. Стало быть, в данном случае разговор о непрерывности морфизмов вполне оправдан, поэтому можно рассматривать в том числе и гомеоморфизмы.

Для объектов более высокого уровня, являющихся комплексами, кроме этого, можно явно моделировать замену компонент, когда заменяющие компоненты вносят свое поведение, отличное от поведения заменяемых. Здесь следует сделать замечание. Выше уже было сказано, что морфизмам подлежат прежде всего множества M и E. Если при заменах компонент нижнего уровня не рассматривать изменений их интерфейсов в комплексах и изменений программ поведения (9), (10), R_9, окажется, что и в комплексе любого уровня за изменения поведения будут отвечать именно замены действий M и реакций E. Казалось бы, отказ от изучения изменений интерфейсов в комплексах и программ поведения компонент существенно снизит общность рассмотрения. Однако, если обратиться, например, к предметной области социальных систем можно видеть, что там дело обстоит именно так: никто из-за вновь прибывшего работника не будет менять сложившиеся в подразделении способы взаимодействия, наоборот, новичку предложат вписаться в существующие. Так же и с программами поведения (9), (10), R_9 – они определяются должностными обязанностями работников, поэтому вместо убывшего врача на работу будет взят тоже врач а не повар, вместо летчика – летчик, вместо шофера шофер и т.д., а вот их отдельные действия и реакции в рамках программ их должностных обязанностей вполне могут различаться.

Морфизмы могут за не слишком продолжительное время привести систему к поведению противоположному исходному, а если они еще и непрерывны – такое изменение поведения может быть замечено далеко не сразу, так как изменения могут быть незаметны на малых временах наблюдения, например, в силу их малости из-за непрерывности морфизмов. Примеров постепенного превращения объектов в нечто совсем иное путем морфизмов много в алгебраической топологии, например чашки в бублик, или человечка со сцепленными руками в человечка с расцепленными [23]. В области социальных наук это Окна Овертона [24]. Их критика некоторыми социологами часто обусловлена отсутствием в распоряжении последних адекватного языка, различающего индивидуальное, групповое и коллективное поведение в сложной системе. Естественный язык эти понятия легко смешивает [25].

Желание поддерживать поведение сложной системы в приемлемых рамках поднимает вопрос об инвариантах, ограничивающих все множество возможных морфизмов. Как работают инварианты? Инвариант, являющийся для сложной системы добавочной аксиомой, выделяет из группы морфизмов подгруппу сохраняющих его допустимых морфизмов. Такая аксиома должна быть дополнительной по отношению к аксиомам рода структуры, так как аксиомы Бурбаковских родов структуры (в том числе, $R_1 - R_{11}$ рода структуры CS) переносимы при изоморфизмах базисных множеств [10], следовательно никак не могут их ограничивать. Такие дополнительные аксиомы-инварианты, ограничивающие преобразования поведения при морфизмах базисных множеств, будем называть идеологическими.

В социуме ограничивающими преобразования поведения инвариантами – идеологическими аксиомами – являются религиозные заповеди, моральные и культурные ценности, традиционные и гражданские нормы поведения, – все что можно назвать Законом с большой буквы. Этот Закон отчасти зафиксирован письменно в текстах культурных и религиозных памятников, юридических кодексов и нормативных документов; отчасти как предание распространяется устно, например, большинство работников успешно исполняют свои должностные обязанности, хотя мало кто из них может вспомнить, когда в последний раз видел их текст. Закон программирует поведение агентов социума в смысле рода структуры CS, т.е. состава базисных множеств M и E, связи событий с действиями (8), R_8 и законченных программ поведения (9), (10), R_9.

Сложная система должна жертвовать часть своей энергии на поддержание законов своего существования в трех планах – физическом, информационном и идейном. Если сложная система в течение определенного времени оказывается неспособной на поддержание своего потенциала динамического равновесия, структуры и аксиом – она перестает существовать в прежнем качестве. Скорее всего, при этом ее базисные множества за бесценок достанутся более успешным системам.

Для поддержания динамического равновесия, сложная система должна, во-первых, поддерживать потенциал динамического равновесия – выполнять определенную минимальную работу в единицу времени (например, вращение не ниже 800 оборотов в минуту для двигателя; оплата зарплат, коммунальных услуг, аренды помещений и т.д., для предприятия). Для поддержания структуры и инвариантов (законов существования) сложной системы должны проводиться некие регулярные мероприятия, которые можно назвать культом, а отдельные его мероприятия - ритуалами [25]. Например, для поддержания в эксплуатации автомобиля, необходимо периодически проходить техобслуживание, техосмотр, заправлять его горючим, менять технологические жидкости. В случае более сложных социальных систем, для сохранения инвариантов поведения приходится периодически программировать их агентов на «приверженность определенным символическим системам», – как формулируют это антропологи [26]. Например, для поддержания науки, как сложной открытой системы необходимы научные ритуалы – проведение конференций, защит дипломов и диссертаций, система научных степеней и званий и правила их получения, заседания Ученых советов, публикация и рецензирование научный работ, издание научных журналов. Важной частью армейского культа, несомненно, является строевая подготовка.

Культ программирует поведение агентов нижнего уровня, позволяющее удерживать коллективное поведение сложной системы в приемлемых рамках, что выражено известным афоризмом П.А. Флоренского: «Культ – основа культуры» [27]. Под зонтичным понятием «культура» в данной работе понимается исключительно ее этологический аспект – способ коллективного поведения сложной системы.

Заключение

Структурная теория 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 крупномасштабных систем – модельный синтез и модельно-ориентированное программирование.

Изучая морфизмы базисных множеств построенной с помощью модельного синтеза системы, и инварианты, ограничивающие такие морфизм, мы получаем математический язык исследования сложных открытых (меняющих свой состав) систем. Выводами, полученными с помощью этого языка, является, например, то, что устойчивое развитие есть modus vivendi сложной открытой системы и что в сложных открытых системах, в отличие от замкнутых физических систем, ведущую роль играет сохранение законов (система жертвует мощность на поддержание своих аксиом и структуры), а не законы сохранения (которые конечно же имеют место).

Геометрическая теория дает математическую языковую среду для дискурса в предметной области моделирования сложных систем, т.е. возможность выявлять достаточно тонкие различия сущностей, которые обычно теряются при их гуманитарном обсуждении на естественном языке.

С помощью языка геометрической теории показано, что для поддержания заданного поведения сложной системы недостаточно одних лишь действий по поддержанию потенциала динамического равновесия. Нужна идейно-информационная система, программирующая сохранение инвариантов, обеспечивающих желательное поведение. В работе она названа культом. Культ наряду с поддержанием потенциала динамического равновесия обеспечивают сохранение законов функционирования сложной системы. Феномен сохранения законов существования сложной системы при ее динамическом равновесии подтверждает мнение П.А. Флоренского, что в основе культуры, как системы общественного поведения, лежит культ [27].

Литература

1. *Платон.* Диалоги. М.: Мысль, 1986. 607 с.
2. *Ершов Ю.Л., Палютин Е.А.* Математическая логика. М.: Наука, 1987. 336 с.
3. *Бор Н.* Атомная физика и человеческое познание. М.: Изд. иностр. лит., 1961. 151 с.
4. *Об основаниях геометрии* // Сборник классических работ по геометрии Лобачевского и развитию ее идей / Ред. и вступ. статья А.П. Нордена, М.: Гос. изд. технико-теоретической литературы, 1956. 533 с.
5. *Дородницын А.А.* Информатика: предмет и задачи // В сб. Кибернетика. Становление информатики. М.: Наука, 1986.
6. *Павловский Ю.Н.* Технологии, объединяющие математические и гуманитарные методы анализа и прогноза сложных процессов (на примере моделей исторических процессов) // В книге Проблемы математической истории: Историческая реконструкция, прогнозирование, методология. / Отв. ред. Г.Г. Малинецкий, А.В. Коротаев. М.: URSS, 2016. С. 104-111.
7. *Бродский Ю.И.* Модельный синтез и модельно-ориентированное программирование М.: ВЦ РАН, 2013. 142 с.
8. *Бусленко Н.П.* Моделирование сложных систем М.: Наука, 1978. 400 с.
9. *Леванский В.А.* Моделирование в социально-правовых вопросах. М.: Наука, 1986, 158 с.
10. *Бурбаки Н.* Теория множеств. М.: Мир. 1965. 456 с.
11. *Маклейн С.* Категории для работающего математика / Перевод с англ. под ред. В.А. Артамонова М.: ФИЗМАТЛИТ, 2004. 352 с.
12. *Теория моделей и алгебраическая геометрия.* // Сб. статей под ред. Э. Бускаран. М.: МЦНМО, 2008. 280 с.
13. *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.
14. *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.
15. *Cohn A., Maréchal M.A., Tannenbaum D., Zünd C.L.* Civic honesty around the globe // Science 05 Jul 2019. Vol. 365, Issue 6448, pp. 70-73. DOI: 10.1126/science.aau8712.
16. *Павловский Ю.Н., Смирнова Т.Г.* Введение в геометрическую теорию декомпозиции. М.: ФАЗИС, ВЦ РАН, 2006. 169 с.
17. *Крон Г.* Исследование сложных систем по частям (диакоптика). М.: Наука, 1972. 544 с.
18. *Бродский Ю.И.* О приближенной декомпозиции модели-компоненты // Моделирование, декомпозиция и оптимизация сложных динамических процессов, 2014. Т. 29, №1(29), С.119-127.
19. *Николис Г., Пригожин И.* Самоорганизация в неравновесных системах. М.: Мир, 1979. 512 с.
20. *Сайт, посвященный С.П. Курдюмову* – http://spkurdyumov.ru (дата обращения: 29.05.2022)
21. *Павловский Ю.Н.* О сохранении структуры вооруженных сил в процессе вооруженной борьбы // Дискретный анализ и исследование операций. Серия 2. 1998. Т. 5. № 1. С. 40-55.
22. *Налимов В.В.* Вероятностная модель языка. М.: Наука, 1979. 303 с.
23. *Мищенко А.С., Фоменко А.Т.* Краткий курс дифференциальной геометрии и топологии. М.: ФИЗМАТЛИТ, 2004. 304 с.
24. *A brief explanation of the Overton window.* // Mackinac Center for Public Policy, official site. https://www.mackinac.org/OvertonWindow (last access – May 29, 2022)
25. *Бродский Ю.И.* Модельный синтез, как подход к геометрической теории поведения // Моделирование, декомпозиция и оптимизация сложных динамических процессов, 2019. Т. 34, №1(34). С. 43-71. DOI:10.14357/24098639190103.
26. *Douglas M.* How Institutions Think. Syracuse (NY): Syracuse University Press, 1986.
27. *Флоренский П.А.* Столп и утверждение истины: Опыт православной теодицеи. М.: АСТ, 2003. 635 с.