

## Анализ исследований математических основ модельно-ориентированной системной инженерии

*В. А. Сухомлин*

Московский государственный университет им. М. В. Ломоносова,  
Российская Федерация, 119991, Москва, Ленинские горы, 1  
Федеральный исследовательский центр «Информатика и управление» Российской академии наук,  
Российская Федерация, 119333, Москва, ул. Вавилова, 44

**Для цитирования:** *Сухомлин В. А.* Анализ исследований математических основ модельно-ориентированной системной инженерии // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. Вып. 3. С. 350–365. <https://doi.org/10.21638/spbu10.2024.304>

В статье дается анализ исследований математических основ модельно-ориентированной системной инженерии (MBSE, Model-Based Systems Engineering). Рассмотрены как классическая математическая теория проектирования систем Уаймора, так и современные исследования, в частности формализации семантики языка SysML с помощью теории автоматов, применения теории категорий в качестве формальной математической базы для системного проектирования на основе моделей, изучение возможностей объединения теоретической основы систем Уаймора и универсального формализма моделирования Discrete Event System Specification (DEVS).

*Ключевые слова:* системная инженерия, модельно-ориентированная системная инженерия, модельно-ориентированная системная инженерия, конечные автоматы, семантика языка SysML, трехдольная теория, моделирование DEVS, теория категорий в модельно-ориентированной системной инженерии.

**1. Введение. Исторические аспекты системной инженерии.** Системная инженерия (system engineering) — научно-прикладная дисциплина, предназначенная для исследования, создания и управления сложными (инженерными) системами. Она основана на системном подходе и объединяет методы, стандарты и инструменты, необходимые для разработки, анализа и управления сложными системами. Системная инженерия играет ключевую роль в обеспечении успешной разработки и эксплуатации современных инженерных систем в различных областях, таких как авиация, космос, оборонное производство, транспорт, энергетика и др.

Многие ученые считают основоположником системного подхода русского философа и писателя А. А. Богданова, который в своей основной работе «Тектология. Всеобщая организационная наука», первый том которой вышел в 1912 г. (см. [1]), попытался обобщить универсальные организационные законы, управляющие поведением и устройством принципиально любых сложных систем, рассматривая организационно-структурные отношения безотносительно к природе субстрата системы, а как общие для физических и биологических, так и для социальных и культурных систем. Тектология Богданова предвосхитила кибернетику Н. Винера [2] и общую теорию систем Л. фон Берталанфи [3].

В 1960-х годах в рамках программ по созданию космических и ракетных систем,

строительству энергетических комплексов сложились основные принципы и методы (а затем и стандарты) системной инженерии, такие как анализ требований, анализ функциональности систем, архитектурное проектирование, верификация, испытания, анализ качества, обеспечение безопасности и т. д. Также были разработаны первые математические модели и инструменты для анализа и оптимизации систем.

Чуть позже были предприняты попытки разработки математической теории систем. Наиболее известными работами в этой сфере были:

- общая теория систем (1975 г. [4]);
- абстрактная теория систем (1989 г. [5]);
- математическая теория системной инженерии (1993 г. [6, 7]).

Более продвинутой в сторону практического применения оказалась математическая теория модельно-ориентированного системного проектирования, разработанная А. Уэйном Уаймором, в основе которой лежало формальное определение модели системы с использованием сложного математического аппарата [6]. Подход к проектированию систем в контексте моделей [7] стал называться методологией системного проектирования на основе моделей, или MBSE (Model-Based Systems Engineering). Термин MBSE Уаймор связывал со своей математической теорией проектирования систем, которую он назвал трехдольной теорией (tricotyledon theory of system design — ТЗСД) [7], или теорией трикотиледонов, что отражало триаду листообразных пространств, в рамках которых осуществляется проектирование моделей систем, а именно пространств функциональности (functionality), возможности построения (buildability) и реализуемости (implementability).

Однако при дальнейшем развитии подхода MBSE большее внимание было обращено на развитие технологий моделирования и управления жизненным циклом систем, но не на базе формального математического аппарата Уаймора, а скорее с оглядкой на него.

В настоящее время, когда подход MBSE стал доминантой в системной инженерии и создание в его рамках современных сверхсложных инженерных систем требует применения высокоавтоматизированных технологий, отсутствие принятой теоретической основы для MBSE становится проблемой для дальнейшего развития средств автоматизации создания систем, в частности, доказуемости их корректности, глобальной оптимизации, а также для достижения концептуальной согласованности используемого инструментария. Например, в работе [8] говорится о том, что модели, разработанные на основном для MBSE языке моделирования систем (SysML), могут иметь множество интерпретаций [9].

Несмотря на то что современные технологии подхода MBSE обеспечивают построение цифровых моделей системы посредством описания элементов системы и связей между ними, таким моделям не хватает формальной базовой основы, которую можно было бы обеспечить с помощью математической теории [10]. В связи с этим проводится большое количество исследований в области математических основ системной инженерии [11–16], что совпадает с позицией Международного совета по системной инженерии (INCOSE), который призвал, что системная инженерия должна базироваться на более строгих основах математики [17].

В данной статье кратко рассматриваются основные результаты наследия Уаймора и некоторых новых исследований в области математических основ системной инженерии.

**2. Котиледоны Уаймора и системная инженерия.** В литературе по теме MBSE, как правило, отмечается основополагающий вклад А. У. Уаймора в разработку

математических основ MBSE [18]. Главные результаты созданной им математической теории для постановки проблемы проектирования системы в контексте, основанном на моделях, изложены в его книгах [6, 7, 19, 20].

Подход, предложенный Уаймором, обеспечивает строгую математическую основу для проектирования моделей систем дискретного времени. Ключевым аспектом его теории, который подробно рассматривается в [19], являются введение теории трехдольных частей T3SD (tricotyledon theory of system design) и определение шести основных категорий требований к проектированию системы (system design requirements — SDR), которые он определяет следующим образом:

$$\text{SDR} = (\text{IOR}, \text{TYR}, \text{PR}, \text{CR}, \text{TR}, \text{STR}),$$

здесь IOR — требование ввода/вывода (I/O requirement), TYR — требование технологий (technology requirement), PR — требование к производительности (performance requirement), CR — требование к стоимости (cost requirement), TR — компромиссное требование (trade-off requirement), STR — требование системного теста (system test requirement).

Название теории T3SD обусловлено тем, что Уаймором термином «семядольные листья» (cotyledons), обозначающих в ботанике структуры зародыша семенного растения, способных образовывать лист после прорастания, стали называться три основных пространства проектирования моделей системы, а именно пространства функциональности (functionality), возможности построения (buildability) и реализуемости (implementability). Тогда окончательный дизайн системы мог бы представляться «цветком» из этих семядолей (котиленонов). Поскольку существуют три основных области проектирования моделей систем, эта теория стала известна как трехдольная теория проектирования систем. Ее применение предполагает, что постановка задачи системного проектирования требует определения трех порядков (критериев), по одному в отношении пространств функциональности (FSR), возможности построения (BSR) и реализуемости (ISR), превращающие их в математические структуры PR, CR и TR соответственно. Также в этой теории описано пространство реализуемых элементов системных тестов, обозначенное ISTISR (implementable system tests items), связанное с требованиями системного теста STR, сгенерированное на основе описанных выше требований IOR и TYR. Визуальное представление ISTISR и его отношения к трем проектным пространствам системных моделей показано на рис. 1.

Ценность основополагающей работы Уаймора и ее применимость к MBSE заключается в предложенных им пространствах FSR, BSR и ISR. Контуры в пределах FSR представляют эквивалентные функциональные конструкции (модели) системы, созданные по требованию (критерию) производительности PR, контуры в пределах BSR — эквивалентные конструкции построенных систем с учетом требований к стоимости CR, а контуры ISR — эквивалентные реализуемые конструкции систем, разработанные на основе требования компромисса TR.

Таким образом, проектирование системы осуществляется через построение ее моделей в трех пространствах (котиленолах) — функциональности, возможности построения и реализации, с учетом требований к производительности PR, к стоимости CR и компромисса TR соответственно. При этом на каждой стадии своего жизненного цикла системы проверяется соблюдение определенных выше шести категорий требований.

**3. Математическая модель системы по Уаймору.** Остановимся несколько подробнее на теории T3SD, в основе которой лежит разработанная Уаймором матема-

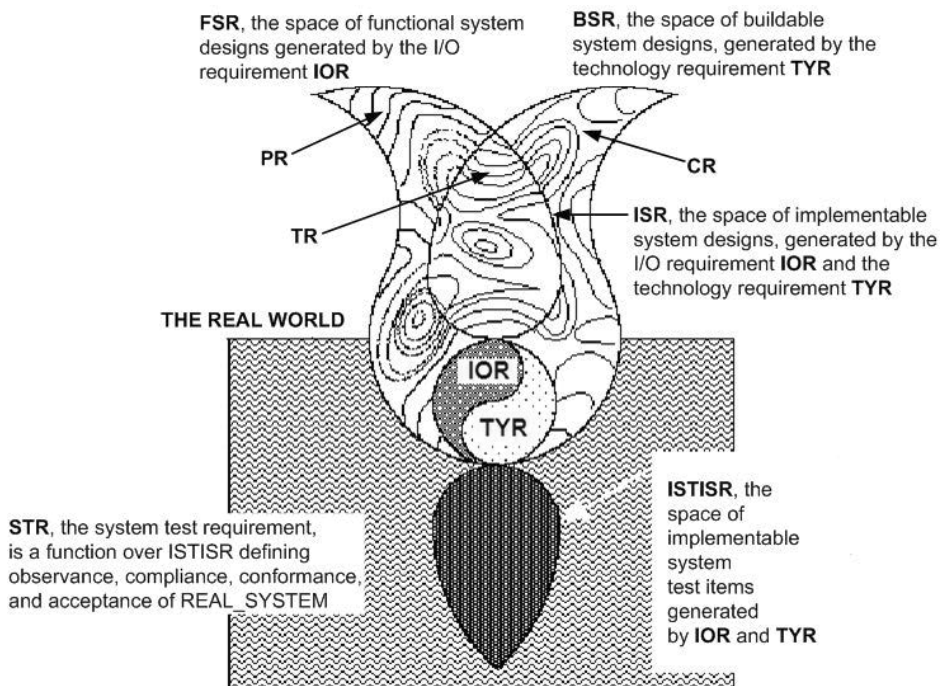


Рис. 1. Требования системного тестирования при привязке проблемы проектирования системы к реальному миру (по [19])

тическая теория систем. Для анализа возьмем теоретико-множественное определение системы, данное в его работе [21], в котором система определяется как набор из семи элементов вида

$$Z = (S, P, F, T, z, Q, \lambda),$$

где  $S$  — набор состояний системы;  $P$  — множество отдельных входов системы;  $F$  — множество временных траекторий входов, допустимых системой;  $T$  — временной масштаб системы;  $z$  — функция перехода состояний системы, такая, что если  $f$  — временная траектория входов системы  $Z$ ,  $f \in F$ ,  $x$  — состояние системы  $Z$  в момент времени  $0$ ,  $x \in S$ , а  $t$  — значение времени во временной шкале  $T$  системы  $Z$ ,  $t \in T$ , тогда  $z(f, x, t)$  — символ состояния системы  $Z$  в момент времени  $t$ , учитывая, что состояние системы  $Z$  в нулевой момент времени равно  $x$  и  $f$  — временная траектория, поставляющая входные данные в систему  $Z$  между моментами времени  $0$  и  $t$ ;  $Q$  — набор отдельных выходов;  $\lambda$  — функция считывания, так что если  $x$  — состояние системы в любой момент времени, то  $\lambda(x)$  — выход системы в данный момент.

Математические артефакты  $S$ ,  $P$ ,  $F$ ,  $T$ ,  $z$ ,  $Q$  и  $\lambda$  также должны удовлетворять некоторым требованиям согласованности [7], которые для нас на этом уровне обсуждения не важны.

На основе такого определения системы описываются различные связанные понятия и отношения между моделями системы, включая входной порт, выходной порт, систему, эксперимент, показатель производительности, подсистему, гомоморфизм, изоморфизм, моделирование, рецепт связи и системный результат рецепта связи.

Особое место в теории Уаймора отводится понятию рецепта связи (Coupling Recipe), под которым понимается описание в математических и системно-теорети-

ческих терминах знакомой блок-схемы компонентов системы, связанных стрелками, указывающими отношения ввода/вывода между компонентами. Результатом рецепта связи является система, компоненты которой приведены на блок-схеме. Системные свойства конечного результата выводятся в рамках этой теории.

На основе введенного определения понятия системы ставится задача проектирования или анализа системы. Она представляет собой 6-кортеж

$$P = (X, T, \alpha, \beta, U, \mu),$$

в котором  $X$  — спецификация ввода/вывода;  $T$  — технология;  $\alpha$  — упорядочивание конструкций системы по критерию производительности в функциональном котиледоне или в пространстве ввода/вывода, т. е. I/O;  $\beta$  — упорядочивание по стоимости в рамках технологического пространства, генерируемой технологиями  $T$ ;  $U$  — упорядочивание в соответствии с требованием компромисса в пространстве осуществимости конструкции системы, генерируемой с помощью  $X$  и  $T$ , которая представляет собой компромисс между  $\alpha$  и  $\beta$ ;  $\mu$  — план тестирования системы.

Спецификация входных/выходных данных представляет собой набор элементов следующего вида:

$$V = (X, G, Y, H, n),$$

где  $X$  — набор отдельных входных данных, для управления которыми существует система, подлежащая проектированию или анализу;  $G$  — набор всех временных траекторий таких входных данных, которыми, возможно, придется управлять разрабатываемой или анализируемой системе;  $Y$  — набор отдельных выходных данных, которые проектируемая или анализируемая система должна производить в результате управления входными данными;  $H$  — набор всех временных траекторий выходных данных, которые может генерировать система, подлежащая проектированию или анализу;  $n$  — функция, сопоставляющая каждой заданной входной траектории набор всех выходных траекторий, которые могут быть созданы в качестве выходных данных любой системой, воспринимающей данную траекторию как входные данные, ограниченные только физическими, биологическими или юридическими ограничениями.

$X$  — спецификация ввода/вывода — это операционное описание целей, ради которых существует система.

Технология  $T$  — любой набор систем, представляющий компоненты, которые можно объединить с помощью рецепта соединения в результирующую систему, способную решить проблему. Технология  $T$  представляет собой средства, доступные для достижения целей.

Система  $Z$  удовлетворяет требованиям  $X$ , если  $Z$  может быть запущена в некотором состоянии  $y$ , так, что если  $Z$  испытывает любую заданную входную траекторию в  $G$ , то  $Z$  создаст в качестве выходной траектории одну из тех, которые находятся в  $H$  и соответствуют входной траектории, реализуемой функцией  $n$ .

Когда спецификация  $X$  определена и тем самым установлено множество конструкций системы в функциональном пространстве, тогда используется оценка  $\alpha$ , чтобы сравнивать любые две системы в этом пространстве по их производительности. Технологическое пространство — это совокупность всех систем, которые можно построить по технологии  $T$ . Система  $Z$  может быть построена по технологии  $T$ , если  $Z$  является результатом рецепта сопряжения (блок-схемы), все компоненты которого находятся в  $T$ . Порядок оценки  $\beta$  определяется таким образом, чтобы сравнить любые две системы в технологическом пространстве относительно использования ресурсов ( $U/R$ ), представленных при их построении и эксплуатации.

Пространство осуществимости — совокупность всех систем, находящихся в функциональном пространстве (ввода/вывода) и реализуемых в данной технологии  $T$ . Система  $Z$  находится в пространстве осуществимости, если она удовлетворяет I/O-спецификации  $X$  и реализуема в технологии  $T$  в том смысле, что есть система  $Z_r$ , которую можно построить в  $T$  (с помощью рецепта связи  $k$ , все компоненты которого находятся в технологии  $T$ ), и она моделирует  $Z$  (под этим понимается, что существует подсистема  $Z_s$  системы  $Z_r$ , гомоморфным образом которой является  $Z$ ). Порядок оценки  $U$  определяется в пространстве осуществимости таким образом, чтобы можно было сравнить любые две системы относительно компромисса между производительностью I/O систем и ресурсоемкостью U/R этих систем в их реализациях.

Идеальным результатом процесса проектирования или анализа системы является выбор системы  $Z_r$  из множества решений в пространстве осуществимости, оптимальной с точки зрения компромиссной оценки  $U$  для реализации  $Z$  в технологии  $T$ , представленной рецептом связи  $k$ , все компоненты которого находятся в технологии  $T$ .

Утверждается, что существенной частью постановки задачи проектирования или анализа любой системы служит способ проверки такой системы  $Z_{real}$ . В плане тестирования системы  $\mu$  должна быть предусмотрена, как часть постановки задачи  $P$ , методология для ответа на вопросы: каковы фактические уровни производительности ввода-вывода и затрат U/R системы  $Z_{real}$ ; является ли  $Z_r$  адекватной моделью  $Z_{real}$ ; является ли система  $Z_{real}$  приемлемой? План тестирования системы  $\mu$ , таким образом, представляет собой практическое, конкретное определение в терминах реального мира абстракций и идеализаций, включенных в другие артефакты постановки задачи  $X$ ,  $T$ ,  $\alpha$ ,  $\beta$  и  $U$ .

В заключение рассмотрим пример, поясняющий применение механизма рецепта связывания [22]. Исходные данные и постановка задачи примера приведены на рис. 2. Требуется построить результирующую систему  $Z^* = \text{RES}(K)$  при соединении  $K$  систем  $Z_i$  для  $i = \overline{1, 4}$  и необходимо указать множество, определяющее связь  $K$ . Пример графического изображения входных/выходных связей дискретных систем  $Z_1, Z_2, Z_3, Z_4$  и как результат связанная система  $Z^* = \text{RES}(K)$  показаны на рис. 2.

В спецификации примера используются исходные обозначения Уаймора; для удобства и ясности добавляются только пробелы ( $\Phi$ ) внутри таблицы с исходными данными и метки за пределами прямоугольников компонент. В нотации Уаймора спецификация связи состоит из объявления всех моделей компонентов системы и для каждой модели входных портов, которые назначаются для связи, а также функций вывода и того, к каким моделям компонентов они подключены.

Наиболее полно математические основы MBSE и описанный выше T3SD-подход Уаймора к разработке систем на основе формальных моделей рассмотрены в его книге [7], которая представляет собой учебник годового курса по теоретическим основам этой дисциплины. Основные цели этого учебника:

- обеспечить теоретические основы систем, необходимые для изучения и освоения практики MBSE;
- объяснить теорию математических систем как основу для разработки моделей и проектов крупномасштабных, сложных систем, состоящих из машин, программного обеспечения и персонала;
- познакомить студентов с теорией проектирования систем T3SD и ее применением к проектированию реальных систем.

Основное содержание книги представлено на языке теории множеств и алгебраи-

Systems	{Input ports}	Input ports which are assigned for coupling	Output functions
$K = \{$	Z1, {P1},	$\Phi, \Phi, \Phi, \Phi,$	$\Phi, \Phi, \xi_{13}, \Phi,$
	Z2, {P2},	$\Phi, \Phi, \Phi, \{P2\},$	$\Phi, \Phi, \xi_{23}, \Phi,$
	Z3, {u32, u31},	$\{u31\}, \{u32\}, \Phi, \Phi,$	$\Phi, \Phi, \Phi, \xi_{34},$
	Z4, {P4},	$\Phi, \Phi, \{P4\}, \Phi,$	$\Phi, \xi_{42}, \Phi, \Phi \}$
		Z1 Z2 Z3 Z4	Z1 Z2 Z3 Z4
		Inputs come from	Output connected to

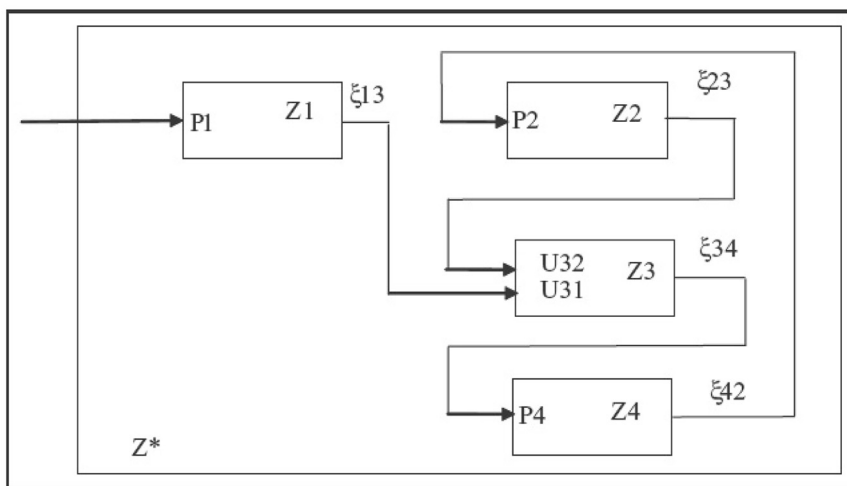


Рис. 2. Пример связанной системы [22]

ческих структур в сочетании с методами формального моделирования, математической логики, теории графов и теории вероятности, применяемых в зависимости от конкретного контекста системной инженерии. К центральным понятиям в ней относится понятие дискретной системы как математической конструкции, состоящей из набора состояний, набора входов, набора выходов, функции следующего состояния и функции считывания (вывода). Также вводятся понятия шкалы времени, входной траектории, траектории состояний, выходной траектории, системных экспериментов. Значительное внимание уделяется математической теории связи систем: как системы могут быть объединены с помощью отношений ввода/вывода для создания иерархических моделей более сложных систем. Ключевыми понятиями для этого являются входные/выходные порты, функций связности, рецепты сопряжения и результаты рецептов сопряжения. Специальная глава посвящена гомоморфным отношениям между системами, использованию понятия изоморфизма системных моделей для упрощения моделей. Вводится концепция системных режимов. Значительная часть книги посвящена формализации ТЗSD-подхода к разработке инженерных систем. В частности, определяется структура постановки задачи проектирования системы, состоящая из шести рассмотренных выше математических артефактов.

Постановка задачи проектирования системы в ТЗSD требует определения трех порядков, по одному на семядоли функциональности, возможности сборки и реализуемости, представляющих математические структуры требований к производитель-

ности, к стоимости и компромисса соответственно. Для формализации процессов жизненного цикла систем вводятся такие понятия как заказы, компромиссные заказы, индексы производительности, конечные функции плотности вероятности, а также показатели качества и их пороги. Еще одним важным понятием является план тестирования реальной спроектированной системы.

**4. Математическая модель системы Уаймора и конечные автоматы языка SysML.** Язык SysML [23, 24] получил широкое распространение для поддержки системного моделирования с точки зрения подхода MBSE. В работе [25] отмечается, что SysML эффективен для визуального представления логической конструкции проектируемых систем, однако он является языком описательного моделирования. Отсутствие базовой математической структуры не позволяет SysML использовать формальную семантику, логику и методы, которые обеспечивают согласованность на протяжении всего развития проекта [26]. В частности, по этой причине известны случаи, когда одна и та же спецификация на SysML имела разные интерпретации при исполнении в различных инструментальных средах.

Успехом в преодолении описательных ограничений SysML могло бы стать определение семантики языка на основе внутренне согласованной математической теории для MBSE, предложенной Уаймором в 1993 г. [7].

В работе [27] проводится исследование возможности введения математической теории Уаймора в семантику языка SysML посредством сравнения работы дискретных систем/автоматов Уаймора с диаграммами конечных автоматов SysML, что представляется актуальным в связи с призывом Международного совета по системной инженерии (INCOSE) к тому, чтобы системная инженерия основалась на более строгом фундаменте математики [18].

Уаймор считал, что его конструкция дискретной системы, весьма близкая к определению автомата Мура, дополненная концепцией рецепта связи (механизмом комплексования систем), способна отобразить любую модель дискретной системы. При этом с помощью теорем и постулатов разработанной Уаймором теории возможен формальный анализ спецификаций моделей дискретных систем. Если это верно и поскольку SysML позволяет создавать модели дискретных систем, то должна существовать хотя бы одна функция, которая преобразует любую модель SysML в математическое определение модели дискретной системы, данное Уаймором. В статье [27] как раз и исследуется верность этой гипотезы с оценкой возможности записи диаграмм конечных автоматов SysML с помощью математических конструкций Уаймора.

Проблема здесь состоит в том, что в языке SysML реализован вариант конечных автоматов (Finite State Machine) в виде диаграмм машин состояний (State Machine Diagram), насыщенный средствами, которые расширяют описательные возможности языка для спецификации поведенческих аспектов систем, по сравнению с классическими конечными автоматами. При этом вводимые расширения не подкреплялись описанием формальной семантики над автоматными конструкциями. Состав основных расширений классической модели конечных автоматов в языке SysML приведен в таблице.

В работе [27] исследуется возможность записи в нотации Уаймора каждой из этих характеристик конечных автоматов SysML, индивидуально и в их комбинации. В результате приводится доказательство возможности отражения в нотации Уаймора характеристик автоматов SysML с 1 по 4 в таблице, а именно состояний, связанных с поведением; поведением, связанным с переходами; поведением, связанным как с состоянием, так и с переходом, и ортогональности. При этом исследование ограни-



Таблица. Описание характеристик конечных автоматов SysML

Характеристика SysML	Описание
1. Действие, связанное с состояниями	Действие, которое выполняется при входе в состояние (например, вывод выходного символа); находясь в состоянии (оператор do), или при выходе из состояния (оператор exit)
2. Действие, связанное с переходами	Действие, которое выполняется во время перехода
3. Действие, связанное как с состояниями, так и с переходами	В этом случае действие связывается как с переходом, так и с состоянием
4. Ортогональность	Состояние, имеющее несколько регионов, каждый из которых характеризуется своим собственным набором вершин, своим набором переходов и действует независимо друг от друга
5. Начальное состояние	Описывает начальное состояние конечного автомата или иерархии для описания истории, вызовов от другого конечного автомата и может использоваться для моделирования
6. Конечное состояние	Описывает конечную точку или точку выхода по стеку вызовов из других конечных автоматов
7. Глубина (иерархия, составные состояния)	Используется для отражения регионов внутри состояния. Позволяет уменьшить масштаб сложного поведения, основанного на состоянии
8. События перехода и меры защиты	События — это входные данные для состояния, которые вызывают переход. Guards — это условие true/false, которое разрешает или предотвращает переход
9. История, действия, задержки, тайм-ауты, условия	Это расширения глубины, вывода и перехода, используемые для уменьшения масштаба конечного автомата

чивалось моделями внешнего поведения системы, т. е. система рассматривалась как черный ящик, состояния которого изменяются под воздействием входных данных, поступающих из окружения системы.

**5. Теория категорий как формальная математическая основа для системного проектирования на основе моделей.** В статье [28] теория категорий представляется как формальная основа для MBSE. Теория категорий — это раздел математики, возникший на основе алгебраической топологии и предназначенный для описания различных структурных концепций из разных математических областей [29]. Ее считают наиболее общей и абстрактной отраслью чистой математики [30]. В отличие от теории множеств теория категорий имеет меньше математических структур, которые управляют трансформацией между объектами, и фокусируется на отношениях между элементами, а не на самих элементах, предлагая чистую теорию функций самих по себе, а не теорию функций, производных от множеств [31].

Теория категорий используется с 1990-х годов в информатике, науке о данных и разработке программного обеспечения (см. [32–40]). Однако использование теории категорий в инженерном моделировании считается новым. Примерами здесь могут служить работы [41] и [42]; в первой показано, что теория категорий может быть адаптирована к современным инструментам и языкам моделирования (например, SysML), а во второй на основе теории категорий исследуются проблемы обеспечения согласованности между моделированием многоуровневых систем (MBSE) и безопасностью (MBSA).

Система представляет собой объект, который позволяет рассматривать различные элементы как единое целое [17]. Таким образом, систему можно легко считать категорией, в которой элементы и компоненты системы являются объектами (вещами) категории, а отношения между элементами и компонентами представлены морфе-

мами (стрелками). Например, заинтересованные стороны, аппаратные компоненты, программные компоненты и системы среды, которые взаимодействуют с интересующей системой, — все это объекты категории, представляющие данную систему [43].

Предложен формальный язык Olog для описания конкретной реальной системы в нотации теории категорий и написания требований. Этот термин введен Д. Спиваком [44] и относится к журналу онтологий, где онтология — это исследование природы данного предмета, а ологи предназначены для записи его результатов. Сам олог можно рассматривать как категорию, объектами которой являются блоки с фразами на естественном языке, а стрелки обозначают отношения между этими блоками. Такие отношения могут быть аспектами, атрибутами или наблюдениями. В ологах коммутативные диаграммы представляют факты. Тип в ологах представляет класс объектов или индивидуумов, обладающих общими характеристиками.

У объекта может быть множество аспектов, например, система имеет миссию, набор требований и может быть разложена на подсистемы (рис. 3). Важно отметить, что аспекты представляют собой функциональные отношения.

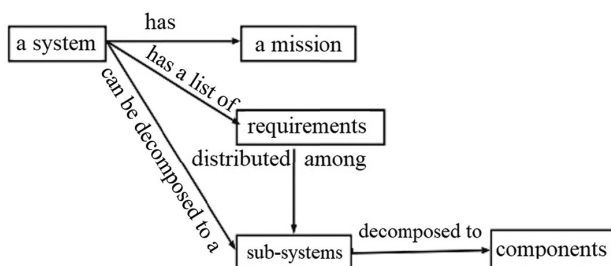


Рис. 3. Множество аспектов объекта

Факты — еще одно понятие в ологах, которое представляет эквивалентность путей. Например, фраза «система имеет список требований, распределенных между подсистемами» эквивалентна фразе «система может быть разложена на подсистемы». Это понятие в теории категорий называется коммутативной диаграммой.

**Система как категория.** Система определяется как набор взаимодействующих элементов, что позволяет рассматривать ее как категорию объектами которой являются все элементы системы [28]. Взаимодействия или отношения между этими объектами могут быть представлены стрелками или морфизмами. Кроме того, поскольку объект сам может быть категорией, систему систем также можно изучать как категорию категорий. С точки зрения проектирования системы требование или набор требований можно считать объектом. Физический элемент системы также является объектом. Отображение или преобразование между требованиями и соответствующими физическими компонентами можно рассматривать как стрелки или морфизмы в категориальном смысле, что облегчит отслеживание системных требований. А концепция эквивалентности путей в теории категорий облегчит описание нескольких альтернатив дизайна.

Атрибуты качества системы, которые используются для оценки производительности системы, иногда называемые «возможностями» системы, такие как доступность, реконфигурируемость и т. д., также могут быть смоделированы как объекты в категории. Связь между такими атрибутами и элементами системы, а также степень удовлетворения каждого атрибута можно описать с помощью стрелок или морфизмов. Для этого определим новую категорию Sys, объектами которой являются

ся элементы интересующей системы. Элемент или объект может быть описательным предложением (например, требованиями или спецификацией системы) или таким элементом как аппаратный компонент, физическая часть или отдельное лицо (например, сотрудник). Проектирование систем на основе моделей — основная часть MBSE. Как правило, процесс проектирования системы представляет собой процесс сопоставления области проблемы с областью решения. Используя язык системного проектирования, процесс проектирования можно рассматривать как преобразование или отображение требований из проблемной области в параметры проектирования и компоненты в области решения. Оно может быть выражено на категориальном языке как морфизмы (функторы) между двумя категориями: категория проблемы (Pro) и категория решения (Sol).

Категория Sol может содержать различные конфигурации или решения, где каждое решение есть результат разных функторов, как показано на рис. 4.

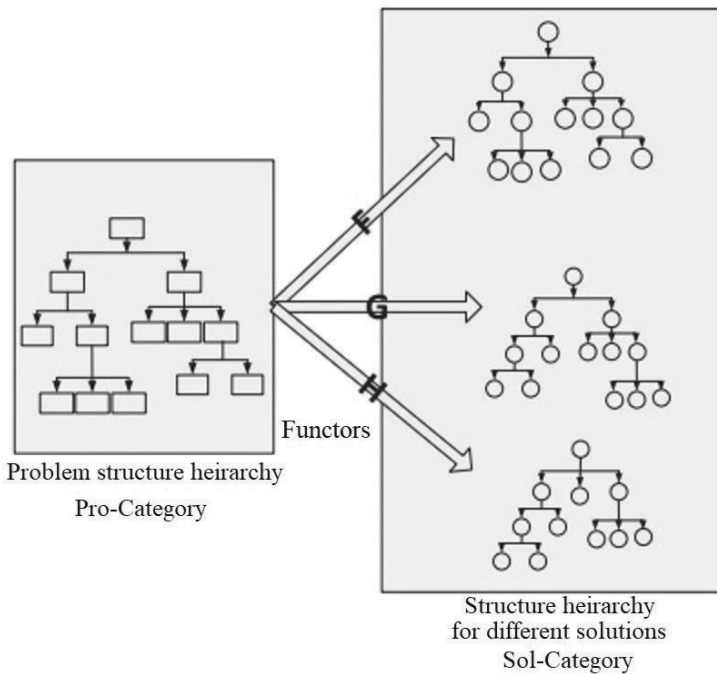


Рис. 4. Преобразование или отображение на категориальном языке как морфизмы (функторы) между двумя категориями: категория проблемы и категория решения [28]

Ологи обеспечивают строгую математическую основу для представления знаний и построения инженерных моделей. Это может оказаться большим подспорьем при формализации MBSE и проектировании на основе моделей.

Свойство эквивалентности путей может быть применено для сравнения различных проектных решений для конкретного требования.

Одним из полезных свойств категорий является принцип двойственности категорий, который учитывает следующее: если свойство  $P$  справедливо для всех категорий, то обратное свойство также справедливо для всех категорий. Этот принцип двойственности можно использовать для проведения валидации и верификации заданной системы.

Заслуживают внимания работы по исследованию применения теории категорий в MBSE [45–47].

**6. Объединение теоретической основы систем Уаймора и формализма моделирования DEVS: к научным основам для MBSE.** В работе [8] разрабатывается метамодель математической теории MBSE Уаймора, что позволяет объяснить ее концепции простыми для понимания терминами и показать их внутреннюю последовательность. Для преодоления ограниченности теории Уаймора предлагается слияние ее теоретической основы с формализмом спецификации дискретной системы событий (DEVS), что открывает новые возможности развития научных основ MBSE.

Цель рассматриваемого исследования — возобновить внимание к теории T3SD, потенциал которой, по мнению авторов [8], не доиспользован, и разработать подход к модернизации этой теории для практического применения в рамках быстро развивающейся цифровой парадигмы проектирования систем; предлагается некоторый упрощенный вариант схемы Уаймора, в котором SysML используется для фиксации ключевых переменных и их взаимосвязей в форме метамодели. Далее на основе исследования T3SD извлечены базовые уравнения этой теории и создана ее метамодель. На основе извлеченных уравнений и метамодели определена область возможных практических приложений T3SD. Далее описывается способ модернизации T3SD посредством слияния с альтернативным формализмом — спецификацией системы DEVS, которая основана на практике и хорошо изучена.

В [19] показано, что DEVS обеспечивает путь к моделированию в подходе MBSE. Кроме того, INCOSE создал совместную рабочую группу с Международной ассоциацией инженерного моделирования и моделирования (NAFEMS), направленную на согласование системного моделирования (MBSE) и моделирования (DEVS) [48]. Так как основа для слияния T3SD и DEVS хорошо обоснована, то формализм DEVS был определен как спецификация подкласса систем Уаймора [6].

В работе [8] обсуждаются подходы и методы слияния формализма Уаймора и DEVS.

**7. Заключение.** Выполненный в статье анализ исследований по формированию математических основ системной инженерии показывает их чрезвычайную востребованность, исходящую из насущной потребности практики в разработке высокоавтоматизированных базирующихся на строгих основах математики технологий создания сложных инженерных систем. По нашему мнению, рассмотренная проблематика вызовет интерес у аспирантов и магистрантов университетов.

## Литература

1. Богданов А. А. Тектология: Всеобщая организационная наука. В 2-х кн. / отв. ред. Л. И. Абалкин. М.: Экономика, 1989. Кн. 1. 304 с.
2. Wiener N. Cybernetics, or communication and control in the animal and the machine. New York: Pfiley, 1948. 231 p.
3. Бертуланфи Л. фон. Общая теория систем — обзор проблем и результатов // Системные исследования. Ежегодник. М.: Наука, 1969. С. 34–35.
4. Mesarovic M., Takahara Y. General systems theory: Mathematical foundations. London: Academic Press, 1975. 268 p.
5. Mesarovic M., Takahara Y. Abstract systems theory. New York: Springer, 1989. 439 p.
6. Wymore A. W. A mathematical theory of systems engineering: the elements. New York: John Wiley & Sons, 1967. 353 p.
7. Wymore A. W. Model-based systems engineering. Boca Raton: CRC Press, Inc., 1993. 710 p.
8. Wach P., Zeigler B. P., Salado A. Conjoining Wymore's systems theoretic framework and the DEVS modeling formalism: toward scientific foundations for MBSE // Appl. Sci. 2021. N 11. P. 4936. <https://doi.org/10.3390/app11114936>

9. Salado A., Wach P. Interpretation discrepancies of SysML state machine: An initial investigation // Proceedings of the 18<sup>th</sup> Annual Conference on Systems Engineering Research (CSER). Redondo Beach, CA, USA, 2020. October 8–10. P. 361–370.
10. Zeigler B. P., Mittal S., Traore M. K. MBSE with/out simulation: State of the art and way forward // Systems. 2018. Vol. 6. N 40. P. 1–18. <https://doi.org/10.3390/systems6040040>
11. INCOSE. *Future of systems engineering (FuSE)*. URL: <https://www.incose.org/about-systems-engineering/fuse> (accessed: June 9, 2020).
12. Rousseau D. The theoretical foundation(s) for systems engineering? Response to Yearworth // Syst. Res. Behav. Sci. 2020. Vol. 37. P. 188–191.
13. NSF. *Workshop: Investigation of the theoretical foundations in systems engineering*. URL: [https://www.nsf.gov/awardsearch/showAward?AWD\\_ID=1548480](https://www.nsf.gov/awardsearch/showAward?AWD_ID=1548480) (accessed: April 9, 2024).
14. NSF. *Workshop: The science of systems engineering*. URL: [https://nsf.gov/awardsearch/showAward?AWD\\_ID=1447031](https://nsf.gov/awardsearch/showAward?AWD_ID=1447031) (accessed: June 9, 2020).
15. Hammami O., Edmonson W. THEFOSE — theoretical foundations of system engineering: A first feedback // Proceedings of the 2015 IEEE International Symposium on Systems Engineering (ISSE). Rome, Italy. 2015. September 28–30. P. 370–374.
16. Collopy P. D. Systems engineering theory: What needs to be done // Proceedings of the 2015 Annual IEEE Systems Conference (SysCon). Vancouver, Canada, 2015. April 13–16. P. 536–541.
17. INCOSE. *INCOSE system engineering vision*. 2025 July. 2014. URL: <https://www.incose.org/docs/default-source/aboutse/se-vision-2025.pdf?sfvrsn=4&sfvrsn=4> (accessed: November 12, 2020).
18. Estefan J. A. Survey of model-based systems engineering (MBSE) methodologies. INCOSE MBSE Focus Group, 2008. [https://www.omgsysml.org/MBSE\\_Methodology\\_Survey\\_RevB.pdf](https://www.omgsysml.org/MBSE_Methodology_Survey_RevB.pdf) (accessed: November 14, 2023).
19. Wymore A. W. Systems movement: autobiographical retrospectives // International Journal of General Systems. 2004. Vol. 33(6). P. 593–610. <https://doi.org/10.1080/03081070412331309131>
20. Wymore A. W. Model-based systems engineering: a quick overview // INCOSE International Workshop. Albuquerque, NM. 2007. January 25.
21. Wymore A. W. Applications of mathematical system theory to system design, modelling and simulation // Winter Simulation Conference Proceedings. 1981. P. 209–219.
22. Ören T., Zeigler B. P. System theoretic foundations of modeling and simulation: a historic perspective and the legacy of a Wayne Wymore // SIMULATION. 2012. Vol. 88. P. 1033–1046.
23. Object Management Group. *System modeling language*. 2016. URL: <http://www.omgsysml.org/> (accessed: December 19, 2016).
24. Friedenthal S., Moore A., Steiner R. A practical guide to SysML. 3<sup>rd</sup> ed. New York: Morgan Kaufman/Elsevier, 2015. 23 p.
25. Vaneman W. K. Enhancing model-based systems engineering with the lifecycle modeling language // 2016 Annual IEEE Systems Conference (SysCon). IEEE. Orlando, FL, USA, 2016. P. 1–7.
26. Graves H., Yvonne B. Using formal methods with SysML in aerospace design and engineering // Annals of Mathematics and Artificial Intelligence. 2011. Vol. 63(1). P. 53–102.
27. Wacha P., Salado A. Can Wymore's mathematical framework underpin SysML? An initial investigation of state machines // Procedia Computer Science. 2019. Vol. 153. P. 242–249. <https://doi.org/10.1016/j.procs.2019.05.076>
28. Mabrok M. A., Ryan M. J. Category theory as a formal mathematical foundation for model-based systems engineering // Appl. Math. Inf. Sci. 2017. Vol. 11. N 1. P. 43–51. <https://doi.org/10.18576/amis/110106>
29. Fokkinga M. M. A gentle introduction to category theory — the calculational approach. Utrecht: University of Utrecht, 1992. 72 p.
30. Hoare C. A. R. Notes on an approach to category theory for computer scientists // Constructive Methods in Computing Science. Berlin; Heidelberg: Springer, 1989. Vol. 55. P. 245–305.
31. Scott D. S. Relating theories of the lambda calculus // To H. B. Curry: Essays on combinatory logic, lambda calculus and formalism. New York: Academic Press, 1980. P. 403–450.
32. Rydeheard D. E., Burstall R. M. Computational category theory. Englewood Cliffs: Prentice Hall, 1988. 257 p.
33. Pierce B. C. Basic category theory for computer scientists. Cambridge: MIT Press, 1991. 114 p.
34. Reed G. M., Roscoe A., Wachter R. F. Topology and category theory in computer science. Oxford: Oxford University Press, 1991. 402 p.
35. Herring J., Egenhofer M. J., Frank A. U. Using category theory to model gis applications // 4<sup>th</sup> International Symposium on Spatial Data Handling. 1990. Vol. 2. P. 820–829.
36. Rising III H., Tabatabai A. Application of category theory and cognitive science to design of semantic descriptions for content data. US Patent 7,319,951. 2008. January 15.

37. Williamson K., Healy M., Barker R. Industrial applications of software synthesis via category theory case studies using specware // *Automated Software Engineering*. 2001. Vol. 8. N 1. P. 7–30.
38. Fiadeiro J. L. Categories for software engineering. Berlin: Springer, 2005. 250 p. <https://doi.org/10.1007/b138249>
39. Kovalyov S. Modeling aspects by category theory // *Proceedings 9<sup>th</sup> Workshop on Foundations of Aspect-Oriented Languages*. Rennes, France, 2010. P. 63–68.
40. Diskin Z., Maibaum T. Category theory and model-driven engineering: from formal semantics to design patterns and beyond // *Proceedings of ACCAT 2012 EPTCS 93*. 2012. P. 1–21.
41. Luzeaux D. A formal foundation of systems engineering // *Complex Systems Design & Management*. Cham: Springer, 2015. P. 133–148.
42. Vidalie J. Category theory for consistency between multilevel system modeling (MBSE) and safety (MBSA). Paris: Universite Paris-Saclay, 2023. 203 p.
43. Atkinson C., Kuhne T. Model-driven development: a metamodeling foundation // *IEEE Software*. 2003. Vol. 20. N 5. P. 36–41.
44. Spivak D. I., Kent R. E. Ologs: a categorical framework for knowledge representation // *PLoS One*. 2012. Vol. 7. N 1. P. e24274.
45. Engel A., Mordecai Y. Systems engineering using category theory // *Systems Science for Engineers and Scholars*. 2024. P. 63–68. <https://doi.org/10.1002/9781394211678.ch18>
46. Breiner S., Subrahmanian E., Sriram R. D. Category theory // *Handbook of Model-Based Systems Engineering*. Cham: Springer, 2022. P. 1–41. <https://doi.org/10.1007/978-3-030-27486-35-1>
47. Myers D. J. Categorical systems — theory. 2023. URL: <http://davidjaz.com/Papers/DynamicalBook.pdf> (accessed: November 14, 2023).
48. NAFEMS: *Systems modeling & simulation working group*. URL: <https://www.nafems.org/community/working-groups/systems-modeling-simulation/> (accessed: April 13, 2021).

Статья поступила в редакцию 12 апреля 2024 г.

Статья принята к печати 25 июня 2024 г.

Контактная информация:

*Сухомлин Владимир Александрович* — д-р техн. наук, проф.;  
<https://orcid.org/0000-0001-9468-7138>, [sukhomlin@mail.ru](mailto:sukhomlin@mail.ru)

## Analysis of research into the mathematical foundations of model-based systems engineering

V. A. Sukhomlin

Lomonosov Moscow State University, 1, Leninskie Gory, Moscow, 119991, Russian Federation

Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, 44, ul. Vavilova, Moscow, 119333, Russian Federation

**For citation:** Sukhomlin V. A. Analysis of research into the mathematical foundations of model-based systems engineering. *Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes*, 2024, vol. 20, iss. 3, pp. 350–365. <https://doi.org/10.21638/spbu10.2024.304> (In Russian)

The article provides an analysis of research into the mathematical foundations of model-based systems engineering (MBSE, Model-Based Systems Engineering). Both the classical mathematical theory of Wymore’s system design and modern research are considered, in particular, the formalization of the semantics of the SysML language using automata theory, the use of category theory as a formal mathematical basis for model-based system design, the study of the possibilities of combining the theoretical basis of Wymore’s systems and the universal Discrete Event System Specification (DEVS) modeling formalism.

*Keywords:* systems engineering, model-based systems engineering, model-based systems engineering, finite state machines, semantics of the SysML language, tripartite theory, DEVS modeling, category theory in model-based systems engineering.

## References

1. Bogdanov A. A. *Tektologiya: Vseobshchaia organizatsionnaia nauka* [Tectology: General organizational science]. In 2 books. Moscow, Ekonomika Publ., 1989, book 1, 304 p. (In Russian)
2. Wiener N. *Cybernetics, or communication and control in the animal and the machine*. New York, Wiley Publ., 1948, 231 p.
3. Bertalanfi L. fon. Obshchaia teoriia sistem — obzor problem i rezul'tatov [General theory of systems — survey of problem and results]. *Sistemnye issledovaniia* [Systematic investigations]. Moscow, Nauka Publ., 1969, pp. 34–35. (In Russian)
4. Mesarovic M., Takahara Y. *General systems theory: Mathematical foundations*. London, Academic Press, 1975, 268 p.
5. Mesarovic M., Takahara Y. *Abstract systems theory*. New York, Springer, 1989, 439 p.
6. Wymore A. W. *A Mathematical theory of systems engineering: The elements*. New York, John Wiley & Sons Publ., 1967, 353 p.
7. Wymore A. W. *Model-based systems engineering*. Boca Raton, CRC Press, Inc., 1993, 710 p.
8. Wach P., Zeigler B. P., Salado A. Conjoining Wymore's systems theoretic framework and the DEVS modeling formalism: toward scientific foundations for MBSE. *Appl. Sci.*, 2021, no. 11, p. 4936. <https://doi.org/10.3390/app11114936>
9. Salado A., Wach P. Interpretation discrepancies of SysML state machine: An initial investigation. *Proceedings of the 18<sup>th</sup> Annual Conference on Systems Engineering Research (CSER)*. Redondo Beach, CA, USA, 2020, October 8–10, pp. 361–370.
10. Zeigler B. P., Mittal S., Traore M. K. MBSE with/out simulation: State of the art and way forward. *Systems*, 2018, vol. 6, no. 40, pp. 1–18. <https://doi.org/10.3390/systems6040040>
11. INCOSE. *Future of systems engineering (FuSE)*. Available at: <https://www.incose.org/about-systems-engineering/fuse> (accessed: June 9, 2020).
12. Rousseau D. The theoretical foundation(s) for systems engineering? Response to Yearworth. *Syst. Res. Behav. Sci.*, 2020, vol. 37, pp. 188–191.
13. NSF. *Workshop: Investigation of the theoretical foundations in systems engineering*. Available at: [https://www.nsf.gov/awardsearch/showAward?AWD\\_ID=1548480](https://www.nsf.gov/awardsearch/showAward?AWD_ID=1548480) (accessed: April 9, 2024).
14. NSF. *Workshop: The science of systems engineering*. Available at: [https://nsf.gov/awardsearch/showAward?AWD\\_ID=1447031](https://nsf.gov/awardsearch/showAward?AWD_ID=1447031) (accessed: June 9, 2020).
15. Hammami O., Edmonson W. THEFOSE — Theoretical foundations of system engineering: A first feedback. *Proceedings of the 2015 IEEE International Symposium on Systems Engineering (ISSE)*. Rome, Italy, 2015, September 28–30, pp. 370–374.
16. Collopy P. D. Systems engineering theory: What needs to be done. *Proceedings of the 2015 Annual IEEE Systems Conference (SysCon)*. Vancouver, Canada, 2015, April 13–16, pp. 536–541.
17. INCOSE. *INCOSE system engineering vision*. 2025, July. 2014. Available at: <https://www.incose.org/docs/default-source/aboutse/se-vision-2025.pdf?sfvrsn=4&sfvrsn=4> (accessed: November 12, 2020).
18. Estefan J. A. *Survey of model-based systems engineering (MBSE) methodologies*. INCOSE MBSE Focus Group, 2008. Available at: [https://www.omgSysML.org/MBSE\\_Methodology\\_Survey\\_RevB.pdf](https://www.omgSysML.org/MBSE_Methodology_Survey_RevB.pdf) (accessed: November 14, 2023).
19. Wymore A. W. Systems movement: autobiographical retrospectives. *International Journal of General Systems*, 2004, vol. 33(6), pp. 593–610. <https://doi.org/10.1080/03081070412331309131>
20. Wymore A. W. Model-based systems engineering: a quick overview. *INCOSE International Workshop*. Albuquerque, NM, 2007, January 25.
21. Wymore A. W. Applications of mathematical system theory to system design, modelling and simulation. *Winter Simulation Conference Proceedings*, 1981, pp. 209–219.
22. Ören T., Zeigler B. P. System theoretic foundations of modeling and simulation: a historic perspective and the legacy of a Wayne Wymore. *SIMULATION*, 2012, vol. 88, pp. 1033–1046.
23. Object Management Group. *System modeling language*. 2016. Available at: <http://www.omgSysML.org/> (accessed: December 19, 2016).
24. Friedenthal S., Moore A., Steiner R. *A practical guide to SysML*. 3<sup>rd</sup> ed. New York, Morgan Kaufman/Elsevier Publ., 2015, 23 p.
25. Vaneman W. K. Enhancing model-based systems engineering with the lifecycle modeling language. *Annual IEEE Systems Conference (SysCon)*. Orlando, FL, USA, 2016, pp. 1–7.
26. Graves H., Yvonne B. Using formal methods with SysML in aerospace design and engineering. *Annals of Mathematics and Artificial Intelligence*, 2011, vol. 63(1), pp. 53–102.
27. Wacha P., Salado A. Can Wymore's mathematical framework underpin SysML? An initial investigation of state machines. *Procedia Computer Science*, 2019, vol. 153, pp. 242–249. <https://doi.org/10.1016/j.procs.2019.05.076>

28. Mabrok M. A., Ryan M. J. Category theory as a formal mathematical foundation for model-based systems engineering. *Appl. Math. Inf. Sci.*, 2017, vol. 11, no. 1, pp. 43–51. <https://doi.org/10.18576/amis/110106>
29. Fokkinga M. M. *A gentle introduction to category theory — the calculational approach*. Utrecht, University of Utrecht Press, 1992, 72 p.
30. Hoare C. A. R. Notes on an approach to category theory for computer scientists. *Constructive Methods in Computing Science*. Berlin, Heidelberg, Springer, 1989, vol. 55, pp. 245–305.
31. Scott D. S. Relating theories of the lambda calculus. *To H. B. Curry: Essays on combinatory logic, lambda calculus and formalism*. New York, Academic Press, 1980, pp. 403–450.
32. Rydeheard D. E., Burstall R. M. *Computational category theory*. Englewood Cliffs, Prentice Hall Press, 1988, 257 p.
33. Pierce B. C. *Basic category theory for computer scientists*. Cambridge, MIT Press, 1991, 114 p.
34. Reed G. M., Roscoe A., Wachter R. F. *Topology and category theory in computer science*. Oxford, Oxford University Press, 1991, 402 p.
35. Herring J., Egenhofer M. J., Frank A. U. Using category theory to model gis applications. *4<sup>th</sup> International Symposium on Spatial Data Handling*, 1990, vol. 2, pp. 820–829.
36. Rising III H., Tabatabai A. *Application of category theory and cognitive science to design of semantic descriptions for content data*. US Patent 7,319,951, 2008, January 15.
37. Williamson K., Healy M., Barker R. Industrial applications of software synthesis via category theory case studies using specware. *Automated Software Engineering*, 2001, vol. 8, no. 1, pp. 7–30.
38. Fiadeiro J. L. *Categories for software engineering*. Berlin, Springer, 2005, 250 p. <https://doi.org/10.1007/b138249>
39. Kovalyov S. Modeling aspects by category theory. *Proceedings of 9<sup>th</sup> Workshop on Foundations of Aspect-Oriented Languages*. Rennes, France, 2010, pp. 63–68.
40. Diskin Z., Maibaum T. Category theory and model-driven engineering: from formal semantics to design patterns and beyond. *Proceedings of ACCAT 2012 EPTCS 93*, 2012, pp. 1–21.
41. Luzeaux D. A formal foundation of systems engineering. *Complex Systems Design & Management*. Cham, Springer, 2015, pp. 133–148.
42. Vidalie J. *Category theory for consistency between multilevel system modeling (MBSE) and safety (MBSA)*. Paris, Universite Paris-Saclay Press, 2023, 203 p.
43. Atkinson C., Kuhne T. Model-driven development: a metamodeling foundation. *IEEE Software*, 2003, vol. 20, no. 5, pp. 36–41.
44. Spivak D. I., Kent R. E. Ologs: a categorical framework for knowledge representation. *PLoS One*, 2012, vol. 7, no. 1, p. e24274.
45. Engel A., Mordecai Y. Systems engineering using Category theory. *Systems Science for Engineers and Scholars*, 2024, pp. 63–68. <https://doi.org/10.1002/9781394211678.ch18>
46. Breiner S., Subrahmanian E., Sriram R. D. Category theory. *Handbook of Model-Based Systems Engineering*. Cham, Springer, 2022, pp. 1–41. [https://doi.org/10.1007/978-3-030-27486-3\\_85-1](https://doi.org/10.1007/978-3-030-27486-3_85-1)
47. Myers D. J. *Categorical systems — theory*. 2023. Available at: <http://davidjaz.com/Papers/DynamicalBook.pdf> (accessed: November 14, 2023).
48. NAFEMS: *Systems modeling & simulation working group*. Available at: <https://www.nafems.org/community/working-groups/systems-modeling-simulation/> (accessed: April 13, 2021).

Received: April 12, 2024.

Accepted: June 25, 2024.

#### Author's information:

Vladimir A. Sukhomlin — Dr. Sci. in Technics, Professor; <https://orcid.org/0000-0001-9468-7138>, [sukhomlin@mail.ru](mailto:sukhomlin@mail.ru)