

На правах рукописи



СТЕПАНОВ

Павел Алексеевич

Модели, алгоритмы и программные средства определения визуальных
языков на основе вычислительных моделей

Специальность 05.13.11 –

Математическое и программное обеспечение вычислительных машин,
комплексов и компьютерных сетей

АВТОРЕФЕРАТ

диссертации на соискание ученой степени

кандидата технических наук

Санкт-Петербург — 2019

Работа выполнена в Федеральном государственном автономном образовательном учреждении высшего образования «Санкт-Петербургский государственный университет аэрокосмического приборостроения»

Научный руководитель: **Охтилев Михаил Юрьевич**
доктор технических наук, профессор, заведующий кафедрой Кафедра компьютерных технологий и программной инженерии СПб ГУАП

Официальные
оппоненты: **Марлей Владимир Евгеньевич**
доктор технических наук, профессор кафедры Вычислительных систем и информатики Федерального государственного бюджетного образовательного учреждения высшего образования «Государственный университет морского и речного флота имени адмирала С.О. Макарова»

Тарасов Анатолий Геннадиевич
доктор технических наук, заместитель начальника кафедры «Автоматизированные системы подготовки и пуска ракет космического назначения» федерального государственного бюджетного военного образовательного учреждения высшего образования «Военно-космическая академия имени А.Ф. Можайского»

Ведущая организация федеральное государственное бюджетное образовательное учреждение высшего образования «Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова».

Защита диссертации состоится "___" _____ 2019 г. в __:__ часов на заседании диссертационного совета Д 002.199.01 при Федеральном государственном бюджетном учреждении науки Санкт-Петербургском институте информатики и автоматизации Российской академии наук (СПИИРАН) по адресу: 199178, Санкт-Петербург, 14-а линия В.О., 39, комн. 401.

С диссертацией можно ознакомиться в библиотеке Федерального государственного бюджетного учреждения науки Санкт-Петербургского института информатики и автоматизации Российской академии наук по адресу: 199178, Санкт-Петербург, 14-а линия В.О., 39 и на сайте <http://www.spiiras.nw.ru/dissovet>

Автореферат разослан «___» _____ 2019 года

Ученый секретарь
диссертационного совета Д.002.199.01
кандидат технических наук



Зайцева А.А.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность. В индустрии информационных технологий машинные языки применяются повсеместно. Их наиболее известным классом являются текстовые языки программирования, для которых существует развитая теория, позволяющая строить трансляторы в машинные команды. В последнее время также широко применяются и визуальные языки, например, язык моделирования UML, диаграммы «сущность-связь», языки моделирования Simulink и LabView. В свете принятого правительством Российской Федерации курса на импортозамещение, крайне важно создать аналоги средств визуального проектирования и разработки, используемых при разработке и эксплуатации сложных технических объектов (СТО), к которым, в первую очередь, можно отнести объекты ракетно-космической отрасли (ракеты космического назначения, разгонные блоки, космические аппараты различного назначения, элементы наземной космической инфраструктуры), объекты топливно-энергетического комплекса (объекты атомной и гидроэнергетики) и пр.

Применительно к вышеназванным СТО, визуальные языки используются не только при конструировании программного обеспечения, но также и при выполнении процедур по оценке их технического состояния (ТС), то есть на всех этапах жизненного цикла СТО. Используемые при этом программно-аппаратные средства далеки от совершенства и не обеспечивают в полной мере эффективного решения названных задач. Достаточно актуальной является задача разработки и сопровождения специального программного обеспечения (СПО), на долю которого приходится большая часть возникающих при проведении оценивания ТС ошибок, неточностей и пр., что значительно снижает результативность рассматриваемых процессов, часто приводит к различным по масштабам авариям и даже катастрофам и в целом не обеспечивает достаточного уровня эффективности применения СТО. При этом важной проблемой является несовместимость математических аппаратов, используемых для разработки средств визуализации и представления информации об эталонном функционировании объекта, что фактически приводит к необходимости работы двух специалистов разной специализации при разработке такого СПО.

Круг задач, решаемых с помощью визуальных языков, не исчерпывается вышеперечисленными. Визуальные языки присутствуют практически во всех отраслях, причем как технических, так и гуманитарных. Все это показывает крайнюю актуальность исследований, направленных на создание методов описания синтаксиса и семантики произвольных визуальных языков и решения прикладных задач с помощью таких описаний.

Степень разработанности темы. Современное состояние предметной области исследования визуальных средств можно охарактеризовать как некоторое насыщение, в котором новые формализмы практически не создаются, зато исследуются свойства и применимость существующих. Фундаментальный вклад в предметную область внесли выдающиеся отечественные и зарубежные ученые, в частности, Ахо А. и Ульман Дж., Вирт Н., Гладкий А.В., Городецкий В.И., Ершов А.П., Котов В.Е., Ляпунов А.А., Майерс Б., Непейвода Н.Н., Терехов А.Н., Тыгу Э.Х., Эхриг Х., Хомский Н., Янов Ю.И. и другие, развившие базовые элементы теории автоматов, теории алгоритмов, теории искусственного интеллекта, математической логики, теории формальных языков и грамматик.

Нужды индустрии, появление и внедрение в практику парадигмы “цифрового общества” привели к развитию разнообразных прикладных теорий,

решающих задачи создания универсальных визуальных редакторов и формализмов, среди которых, в частности, можно отметить

- 1) схемы представления алгоритмов анализа информации на основе G-сетей, научная школа проф. Охтилева М.Ю. и проф. Соколова Б.В., СПИИРАН, Россия
- 2) многоагентные, иерархические, P2P модели, научная школа Городецкого В.И., СПИИРАН, Россия;
- 3) формализм алгоритмических сетей, научная школа проф. Иванищева В.В., проф. Морозова В.П. и проф. Марлея В.Е; СПИИРАН, Россия;
- 4) технологию REAL и генератор графических редакторов QReal, научная школа проф. Терехова А.Н., Санкт-Петербургский Государственный Университет;
- 5) формализм расширенных позиционных грамматик, научная школа проф. Дж. Костаглиолы, университет Салерно, Италия;
- 6) формализм грамматик упорядочивания изображений, разработанный в Университете Брауна, Род Айленд, США;
- 7) формализм графических функциональных грамматик К. Коджимы и Б. Майерса, Университет Джона Монаша, Австралия;
- 8) различные решения производителей средств CASE и RAD, предназначенные для расширения возможностей их графических редакторов пользователем, в частности, Eclipse GMF и Microsoft DSL Tool.

Цель исследования – сокращение сроков и трудоемкости проектирования программных средств наглядного отображения и оценивания технического состояния сложных технических объектов с использованием визуальных языковых моделей предметной области.

Объектом исследования являются визуальные языки, используемые при проектировании прикладного ПО наглядного отображения и оценивания ТС СТО, **предметом исследования** являются модели, методы и алгоритмы, а также разработанные на их основе программные средства, предназначенные для визуализации и оценивания ТС СТО и поддержки жизненного цикла прикладного ПО.

Основные задачи. В рамках проводимых диссертационных исследований были поставлены следующие задачи:

- 1) провести анализ проблемных аспектов существующих моделей и методов формализации визуальных языков;
- 2) разработать модели и алгоритмы, обеспечивающие поддержку синтаксиса и семантики визуальных языков, доступные широкому кругу лиц;
- 3) разработать универсальный комплекс программ, реализующий отмеченные алгоритмы и модели;
- 4) исследовать применимость моделей, алгоритмов и разработанного комплекса программ средств разработки программного обеспечения (ПО) для задач проектирования и визуального контроля технического состояния СТО.

Научная новизна диссертационной работы состоит в следующем:

- 1) разработан подход к созданию редакторов диаграмм, используемых в задачах разработки и контроля технического состояния сложных технических объектов, основанный на описании визуального языка пользователем, отличающийся от аналогичных подходов тем, что объединяет описание предметной области и описание визуализации в единую математическую модель;
- 2) разработаны модели и алгоритмы, обеспечивающие поддержку синтаксиса и семантики визуальных языков, доступные широкому кругу лиц, отличающиеся

тем, что за основу описания синтаксиса визуального языка взята вычислительная модель Тыугу;

- 3) разработан программный комплекс - универсальный редактор диаграмм, для работы с вышеописанными моделями и алгоритмами, отличающийся использованием для описания синтаксиса визуального языка вычислительных моделей;
- 4) для языка диаграмм “сущность-связь” разработано описание синтаксиса, отличающееся тем, что за основу описания взята вычислительная модель, и продемонстрирована возможность создания диаграмм и их трансляции в сценарии создания схем баз данных, а также для мнемосхемы тракта наддува топливных баков ракеты “Союз-2”, разработано описание, отличающееся тем, что за основу описания взята вычислительная модель, и продемонстрирована возможность использования полученной мнемосхемы при контроле технического состояния системы.

Методология и методы исследования. При выполнении диссертационных исследований использовались элементы теории искусственного интеллекта, теории множеств, теории графов, теории категорий, теории алгебраических графовых трансформаций и теории формальных языков, а также математической логики.

Теоретическая значимость полученных в диссертационной работе результатов заключается в получении единой математической модели описания функционирования и графического представления сложного технического объекта, при этом имеется возможность определения правил визуализации ограниченным математическим аппаратом, а также возможность преобразования в различные артефакты.

Практическая значимость полученных результатов состоит в возможности внедрения в модель СТО правил ее визуализации, что приводит к возможности ее использования не профессиональными программистами, а экспертами в области контроля качества и эксплуатации СТО. За счет этого стоимость разработки ПО автоматизированных систем оценки качества СТО снижается на 20-30%, а время разработки на 10-15%.

В целом, все полученные в работе результаты направлены на формирование единого системного подхода к разработке систем визуализации информации, характерных для телеметрических систем передачи, приема, преобразования и обработки информации.

Основные положения, выносимые на защиту:

- 1) Расширенная вычислительная модель визуального языка для поддержки человеко-машинных интерфейсов.
- 2) Алгоритмы расчета вычислительной модели визуального языка и приведения ее к стационарному виду.
- 3) Метод трансляции диаграмм, заданных в вычислительной модели визуального языка, в программы на других языках.
- 4) Метод построения визуальных интерфейсов диаграмм, заданных в вычислительной модели визуального языка, позволяющий, в частности, построение диалогов с пользователем и сопряжение с телеметрической информацией.

Реализация. Основные теоретические положения и практические результаты работы были использованы при выполнении работ по гранту РФФИ 00-07-90344 “Разработка метамоделей, методов, инструментальных средств и технологии

конвертации проектов информационных систем, созданных в соответствии с различными методологиями в различных CASE-системах”, при проектировании ПО автоматизированной системы управления подготовки к пуску ракеты-носителя Союз-2 в АО “РКЦ Прогресс”, АО “СКБ Орион”, а также в учебном процессе в Санкт-Петербургском Государственном университете аэрокосмического приборостроения, при чтении курсов по теории искусственного интеллекта и функциональному и логическому программированию.

Обоснованность и достоверность сформулированных научных положений, основных выводов и результатов обеспечивается за счет анализа состояния исследований в данной области, согласованности теоретических выводов с результатами экспериментальной проверки разработанных алгоритмов в ходе испытаний и штатной эксплуатации семейства ракет-носителей типа «Союз-2».

Публикации и апробация. Основные положения и результаты диссертационных исследований опубликованы в 18 печатных работах, в том числе 1 статье в издании, индексируемом Scopus/WoS, 6 статьях в статьях в рецензируемых научных изданиях, входящих в Перечень Высшей Аттестационной комиссии, текстах 9 докладов на всероссийских и ведомственных научно-технических конференциях и семинарах, 3 свидетельствах о государственной регистрации программ для ЭВМ. Основные результаты диссертационной работы докладывались и обсуждались на 9 научных конференциях, в том числе пяти международных.

Личный вклад автора составляют теоретические выводы и практические решения, результаты тестирования. Основные научные положения сформулированы и изложены автором самостоятельно

Структура и объем диссертации. Работа состоит из введения, четырех разделов, заключения, библиографического списка (274 наименования) и шести приложений. Объем основной части работы составляет 110 страниц машинописного текста.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении обоснована актуальность проводимых исследований, определены цели работы, основные задачи исследования, отражена научная новизна и практическая значимость работы, сформулированы результаты, выносимые на защиту.

В первой главе дается обоснование преимуществ визуальных языков над текстовыми, анализируются и систематизируются существующие способы формальных описаний визуальных языков, проводится формальная постановка задач исследования.

Кроме наглядности результата и возможности быстрого вмешательства при некорректной работе оцениваемого объекта системы графической визуализации обладают и недостатками, главным из которых является необходимость программирования при их разработке. Проблема может быть устранена путем создания библиотек стандартных правил визуализации, например, в случае ракетно-космической техники (РКТ) в такую библиотеку могут входить топливные и воздушные баки, трубопроводы, вентили и другие элементы РКТ. Но появление нового или уточнение существующего правила визуализации приводит к необходимости переработки программного кода. Так как контролем ТС занимаются инженеры другой специализации, нежели программирование, это приводит к необходимости найма персонала для поддержки соответствующего СПО. Таким образом, возникает задача ухода от программирования при визуализации ТС оцениваемого объекта.

Решением задачи может являться объединение модели функционирования объекта и модели его визуального представления. Описав правила визуализации объекта тем же математическим аппаратом, что и его модель, можно добиться упрощения модификации правил визуализации и избавиться от штата программистов. Более того, на основе этого подхода могут быть реализованы визуальные инструменты, используемые при проектировании программных систем. Таким образом, возникает задача выбора математического аппарата, для описания а) математической модели эталонного функционирования оцениваемых объектов, б) их визуального представления.

Для создания метода описания визуального языка необходимо построить его формальную модель. Определим диаграмму как структурированную систему связанных изображений. Она состоит из сущностей двух видов – изображений и отношений между ними, и математически выражается как: $D = \{I, R\}$, где D – диаграмма; I – множество изображений; R – множество отношений между этими изображениями. I может содержать в себе произвольные изображения, в то время как для отношений определены основные категории: отношения позиционного, размерного, временного синтаксиса и синтаксиса правил. Дополнительно введем понятия синтаксиса взаимодействия и структуры, предназначенные для описания взаимодействия диаграммы с пользователем через редактор. Таким образом $R = \{R_p, R_d, R_t, R_r, R_i, R_s\}$, где R_p – отношения позиционного синтаксиса, R_d – отношения размерного синтаксиса, R_t – отношения временного синтаксиса и R_r – отношения синтаксиса правил R_i – отношения синтаксиса взаимодействия R_s – отношения синтаксиса структуры.

Базовым определением визуального языка является его определение как множества всех допустимых для него диаграмм. Так как перечислить их все невозможно, существует ряд подходов к формализации: подходы на основе графовых грамматик, теории категорий, лямбда-исчисления, искусственного интеллекта и объектов с шаблонами поведения.

Графовые грамматики (ГГ) являются обобщением текстовых языков на визуальные языки. Визуальный язык описывается через понятие визуального алфавита и продукционных правил. Графовой грамматикой называется множество $VL = \{VT, VN, VP, \varepsilon\}$, где VT – множество визуальных терминалов или визуальный алфавит (множество всех элементарных изображений), VN – множество всех нетерминалов, VP – множество продукционных правил, ε – начальный граф языка.

У ГГ существует ряд специфических недостатков, а именно:

- У графа отсутствует начало и конец, нет единой точки для начала анализа
- Граф многомерен, алгоритмы имеют большую вычислительную сложность
- Взаиморасположение узлов графа на холсте может иметь значение.
- Семантическая основа графа является его сравнительно малым подмножеством, в связи с чем принято разделять диаграмму на абстрактный граф синтаксиса (АГС), содержащий в себе смысл диаграммы, и пространственный граф отношений (ПГО), содержащий визуальные элементы, предназначенные для человека, с последующим игнорированием последнего. Таким образом, исследователи игнорируют существенную часть визуального синтаксиса.

Все это существенно усложняет формализм ГГ по сравнению с текстовыми языками. Тем не менее, подходы с использованием ГГ имеют в своей основе проработанные формализмы и для них адаптированы основные алгоритмы.

Подход на основе теории категорий является обобщением ГГ с помощью теории более высокого уровня. Основным представителем является теория алгебраических графовых трансформаций (АГТ), существующая в двух вариантах – на основе одинарного и двойного кодекартова квадрата. Разница между двумя подходами заключается в том, можно ли выполнять операции добавления и удаления вершин в графе одним морфизмом или нет.

Подход на основе теории категорий является наиболее общим и универсальным подходом. Одновременно он крайне сложен – подход оперирует графами, в которых возможны висящие дуги и дуги между другими дугами. Преобразование визуальных языков в такие графы является отдельной проблемой.

Подход на основе λ -исчисления трактует диаграмму через применение узлов и ребер у начального пустого графа. Наиболее проработанным представителем является индуктивный граф, первоначально предназначенный для описания синтаксиса визуального языка VEX. Формализм обладает серьезными преимуществами при необходимости формальных доказательств свойств языка, однако в виду того, что графовые трансформации предполагают только добавление узлов, его выразительные свойства весьма ограничены.

Подходы на основе искусственного интеллекта (ИИ) предполагают обучение графических редакторов на основе шаблонов, выбранных разработчиком. Редакторы при этом ориентируются на разнообразные методы ИИ – нечеткую логику, генетические алгоритмы, семантические сети в попытке генерировать по шаблонам такой редактор, который будет удовлетворять поставленной задаче. Подход является преимущественно интерфейсным и требует понимания работы нижележащих алгоритмов для правильного выбора шаблонов для обучения.

Подход на основе объектов с шаблонным поведением используется в простых редакторах схем. В нем создается библиотека объектов с предопределенным поведением, но крайне ограничен синтаксис, связанный с отношениями между объектами. Сильной стороной подхода является удобство создания иллюстраций, а главный недостаток – отсутствие проверки синтаксиса, ограниченное взаимодействие с пользователем и невозможность трансляции.

Таким образом, можно сделать вывод, что существующие формализмы либо обладают широкими выразительными свойствами, но крайне сложны в прикладном использовании, либо просты, но не обладают нужной гибкостью.

Для представления знаний о штатном функционировании оцениваемого объекта часто используется **вычислительная модель Тыгу**. Она представляет собой формализованную семантическую сеть, в узлах которой находятся параметры системы и формулы, а связи представляют входные параметры и выходные значения формул. Модель в первую очередь ориентирована на вычислительные задачи, но будучи дополнена визуальной составляющей, может быть применена к описанию визуального языка. В этом случае эксперт, модифицирующий СПО для визуализации ТС оцениваемого объекта, может формулировать изменения в математической модели поведения объекта и в средствах его визуализации пользуясь единой моделью. Таким образом, возникают подзадачи расширения модели визуальной составляющей, реализации транслятора входного языка модели и средства визуализации на основе модели.

Вычислительная модель Тыгу нестационарна, то есть вычислительные процессы могут идти разными путями и давать разные результаты. В системах взаимодействия с человеком такой подход применяться не может, так как оператор

СПО не ожидает получить множество диаграмм, из которого будет необходимо выбрать одну. Так возникает подзадача приведения модифицированной вычислительной модели к стационарному виду (в своей визуальной части).

Еще одним важным вопросом является получение новых артефактов на основе взаимодействия со средством визуализации. Например, используя средства визуального проектирования программного обеспечения, оператор ожидает получить на выходе шаблоны кода, а при проектировании баз данных на выходе должен быть сценарий на языке SQL. Отсюда возникает подзадача трансляции вычислительной модели в артефакты предметной области.

Подытоживая вышесказанное, создание или модификация нового визуального средства контроля может быть представлена схемой (рис. 1):

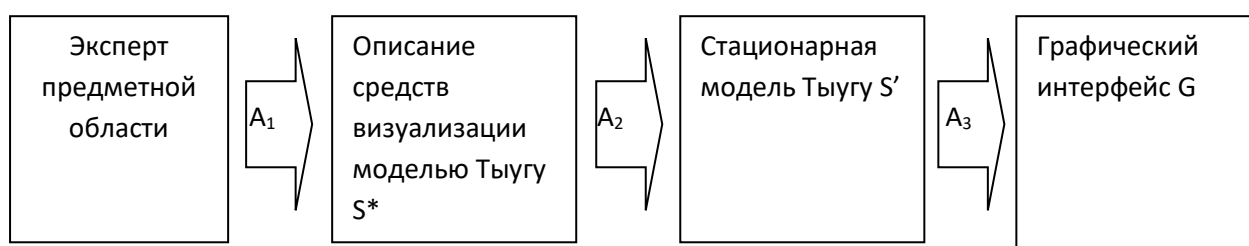


Рис. 1. Схема решения задачи.

Где A_1 – средства представления правил визуализации экспертом предметной области; S^* – модель Тыугу, адаптированная для визуальных языков, полученная из A_1 ; A_2 – алгоритм преобразования модели в стационарную модель; S' – стационарная модель Тыугу, адаптированная для визуальных языков; A_3 – алгоритм преобразования модели в графический интерфейс; G – графический интерфейс, визуальное представление оцениваемого объекта. Для получения артефактов при взаимодействии с графическим интерфейсом G необходимо добавить A_4 – метод трансляции модели и D – результат трансляции (артефакт). Таким образом, для решения задачи объединения знаний о штатном поведении оцениваемого объекта и правил его визуализации необходимо последовательно описать A_1 , S^* , A_2 , S' , A_3 , G , A_4 и D .

Во второй главе вводится вычислительная модель визуального языка, показывается ее применимость к задачам визуализации диаграмм, вводятся алгоритмы для вычисления модели и приведения модели в стационарный вид, демонстрируется метод описания визуального языка на основе модели.

Для решения поставленных задач опишем визуальный язык как набор значений координат и атрибутов объектов, их графических свойств и соотношений между этими значениями. Формализовав объекты, отношения и правила их обработки, можно получить адаптированную вычислительную модель диаграммы.

Назовем атрибутом пару $a = \{n, v\}$, где n – имя атрибута или $\emptyset$, если оно не задано, v – значение атрибута или $\emptyset$, если значение не определено.

Назовем визуальным примитивом $v = \{A, f_g(A)\}$, где A – множество атрибутов примитива, $f_g(A)$ – графическая функция (функция, преобразующая значения атрибутов в изображение).

Назовем диаграммой множество $D = \{A, G, V, R\}$, где A – множество атрибутов, такое, что $A = A_n \cup A_v$, где A_n – множество не визуальных атрибутов и A_v – множество визуальных атрибутов, то есть существует однозначное отображение из A_v в V , G – алгебраическая система, заданная на множестве атрибутов, V – множество

визуальных элементов диаграммы, $v \in V \vdash v \in V_T$, где V_T – предопределенное множество известных визуальных элементов, для которого верно, что $\forall a \in v, v \in V \vdash a \in A_v, \forall a \in A_v \vdash a \in v \in V$, т.е. множество A_v состоит из тех и только тех атрибутов, которые принадлежат визуальным элементам из V , R – множество правил вида $cond(c_i): a_i = f_j(A); a_i \in A$. f_j задано в рамках алгебраической системы G , где c_i – условие, при истинности которого выполняется присвоение $a_i = f_j(A)$ (для краткости в случае, если условие всегда истинно, правило записывается как $a_i = f_j(A)$).

Важной особенностью вычислительной модели является недетерминированность ее вычисления. Необходимо предложить правила вычисления семантической сети такие, чтобы они позволяли при правильном проектировании фрагментов сети детерминированное вычисление диаграммы.

Рассмотрим пример фрагмента диаграммы, имеющий место для большинства диаграммных языков. Пусть задан прямоугольный блок, над которым можно производить следующие действия – менять его размер, перемещая углы блока. Такой фрагмент может быть описан следующей семантической сетью (рис.2):

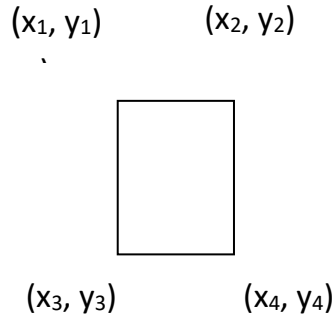


Рис.2 Пример типовой фигуры

Фигура с рис.2 описывается системой из двадцати уравнений:

$$\begin{array}{lllll}
 x_1 = x_3, (1) & y_1 = y_2, (5) & x_2 = x_1 + w, (9) & y_3 = y_1 + h, (13) & w = x_2 - x_1, (17) \\
 x_3 = x_1, (2) & y_2 = y_1, (6) & x_1 = x_2 - w, (10) & y_1 = y_3 - h, (14) & w = x_4 - x_3, (18) \\
 x_2 = x_4, (3) & y_3 = y_4, (7) & x_4 = x_3 + w, (11) & y_4 = y_2 + h, (15) & h = y_3 - y_1, (19) \\
 x_4 = x_2, (4) & y_4 = y_3, (8) & x_3 = x_4 - w, (12) & y_2 = y_4 - h, (16) & h = y_4 - y_2, (20)
 \end{array}$$

где w – ширина, h – высота, а x_1 - x_4 и y_1 - y_4 – атрибуты фигуры с рис.2.

Предположим, пользователь передвинул по оси абсцисс левый верхний угол прямоугольника. Существуют различные пути пересчета модели, потенциально дающие различные результаты. Например, вычисление может пойти по пути $(17) \rightarrow (11)$, или по пути $(2) \rightarrow (9)$; в первом случае фигура изменит свой размер, а во втором – нет. Таким образом, необходимы детерминированные правила вычисления получившейся семантической сети.

Определим воздействие на диаграмму $D = \{A, G, V, R\}$ как подмножество множества $A^* = \{a_1^*, a_2^*, \dots, a_n^*\}$ где для любого $a_i = \{n_i, v_i\} \in A$ $a_i^* = \{n_i, v_i^*\}$ т.е. атрибуты с совпадающими именами имеют, возможно, различные значения.

Алгоритм 1 (алгоритм вычисления диаграммы при воздействии). Вычисление состоит из двух этапов. На первом этапе осуществляется первоначальное изменение переменных (воздействие). На втором этапе применяются правила до тех пор, пока не обнаруживается, что больше не осталось применимых правил. Любое правило из множества R может быть выполнено при соблюдении следующих условий:

- 1) v_i еще не получило новое значение;
- 2) $v_n \dots v_m$ содержат хотя бы одну переменную, получившую новое значение;
- 3) Условие *cond* истинно.

Назовем детерминированным вычисление диаграммы, при котором алгоритм 1 независимо от воздействия генерирует только одну последовательность применения правил.

Назовем непротиворечивым (стационарным) вычисление диаграммы, при котором алгоритм 1 независимо от воздействия приводит к одному и тому же состоянию диаграммы.

Алгоритм 2 (алгоритм приведения к детерминированному виду)

Пусть даны три функции:

$P_i(a)$, возвращающая истину, если x принадлежит воздействию и ложь в противном случае;

$P(a)$, возвращающая истину, если x уже получил новое значение при выполнении алгоритма 2.1 и ложь в противном случае и

$F(a_1, a_2, \dots a_i)$, возвращающая для каждого возможного воздействия I_j последовательность вычисления узлов $n_j = \{a_{n_j,1}, a_{n_j,2}, \dots a_{n_j,k_j}\} \in N$, где N - множество всех последовательностей, k_j - длина последовательности n_j , а $a_{n_j,m}$ - атрибут, вычисляемый на шаге m этой последовательности. Для простоты можно считать, что функция возвращает индекс j последовательности из множества N . Эта функция может быть преобразована в $F'(a_1, a_2, \dots a_i, j) \equiv F(a_1, a_2, \dots a_i) = j$, а F' уже выражена через $P_i(a)$ и функции математической логики.

Пусть задано множество правил $R' = \{r_{n_j,m}\}$, где $r_{n_j,m}$ - правило, которое должно быть применено для вычисления атрибута $a_{n_j,m}$ на шаге m последовательности n_j в соответствии с условиями алгоритма 1. Для каждого значения n_j выполним следующие действия:

Введем в диаграмму атрибуты $a'_{n_j,0}, a'_{n_j,1}, \dots a'_{n_j,k_j}$ логического типа

Каждое правило $r_{n_j,m} : \text{cond}(c_{n_j,m}) : a_{n_j,m} = f_{n_j,m}(A); a_{n_j,m} \in A, m \in 1..k_j$ модифицируем в форму $r_{n_j,m} : \text{cond}(c_{n_j,m} \wedge F'(a_1, a_2, \dots a_i, j) \wedge \bigwedge_{i=1}^{m-1} P(a'_{n_j,i})) : a_{n_j,m} = f_{n_j,m}(A); a_{n_j,m} \in A, m \in 1..k_j$

В результате выполнения алгоритма будет получена новая вычислительная модель, выполнение которой будет происходить детерминированно.

Пользователь в процессе работы оперирует объектами и правилами трансформации диаграммы. Поэтому язык должен быть описан на основе перечня объектов и правил трансформации, заданных в вычислительной модели. Это описание строится на основе описания фрагментов семантической сети и правил их вставки в диаграмму. При этом можно поддерживать как синтаксис самих объектов, так и специальное поведение диаграммы в ответ на изменение топологии.

В третьей главе дается описание прикладной реализации вычислительной модели визуального языка и программного средства для ее поддержки.

Вычислительная модель визуального языка сформулирована в общем виде. Для создания инструментального средства необходимо выбрать конкретные типы данных и операций, при этом получившееся программное средство не должно накладывать существенные ограничения на визуальные языки, поддержка которых является частью оригинальной задачи. Таким образом, необходимо определить:

- Набор поддерживаемых типов данных
- Перечень поддерживаемых математических операций и встроенных функций, используемых для установления соотношений
- Перечень и семантику визуальных примитивов

Прикладная вычислительная модель, использованная в программном средстве, имеет слабую типизацию. Решение о конкретном типе данных результата принимается в процессе вычислений.

Типы данных могут быть трех базовых типов – атомарного, спискового и специального типа “объект”. Кроме того, на основе спискового типа строятся синтетические типы “множество”, и “отображение”.

В качестве атомарных типов данных выбраны следующие типы: логический (Boolean), строковый (String) и числовой (Number). Также вводится тип “список” – упорядоченный набор элементов произвольного типа. Список может содержать в себе другие списки и элементы различного типа. Очевидным способом список может быть расширен до синтетических типов “множество” и “отображение”.

Также вводится тип данных “объект”, предназначением которого является установление отношения принадлежности фрагментов сети логическим сущностям. Необходимость объектного типа проистекает из задачи удаления логически связанных фрагментов из диаграммы. Объекты указывают на иерархию узлов, которые им принадлежат, и могут быть обработаны как единое целое.

Для атомарных типов данных вводятся два типа операций – арифметические и логические, а также набор функций. Для списковых типов дополнительно вводятся списковые операции. Тип объект может участвовать в операциях установления отношений принадлежности, а также удаления. Ряд специальных функций предназначены для анализа свойств атрибутов, модификации структуры модели и организации пользовательского интерфейса.

В результате моделирования было показано, что приведенный перечень типов данных достаточен для реализации поставленных задач.

Визуальный алфавит прикладной вычислительной модели состоит из точки приема (Accept Point), и активной области (Active Area), обрабатывающих события от интерфейса, круга (Circle), эллипса (Ellipsis), линии (Line), текста (Text). В дальнейшем этот набор был расширен понятиями ломаной линии и контура.

Математическое ядро редактора диаграмм представляет собой программную реализацию прикладной вычислительной модели. Ядро состоит из множества атрибутов, транслятора выражений, графа зависимостей, описывающего структуру сети, и встроенного набора визуальных примитивов.

Множество атрибутов представляет собой множество пар вида внутреннее имя: значение. Каждый атрибут в модели имеет уникальное внутреннее имя. При помещении объекта в модель происходит преобразование имен атрибутов в уникальные идентификаторы. Таким образом, если объект будет помещен в модель дважды, модели появятся два разных объекта с одинаковой внутренней структурой,

но с уникальными именами атрибутов. При помещении в модель нового соотношения между атрибутами также выполняется преобразование имен, чтобы отношение было установлено именно на тех уникальных именах атрибутов, которым соответствуют атрибуты, использованные в отношении.

Аналогично устанавливаются связи между атрибутами разных объектов. В отношении атрибуты поступают в виде псевдонимов, для которых известно, какие внутренние имена они представляют; эти псевдонимы преобразуются во внутренние имена и отношения добавляются в вычислительную модель.

Особенностью модели является транзакционность. Транзакция поддерживает три этапа – подготовка, выполнение и фиксация либо откат. На этапе подготовки все измененные атрибуты считаются частью воздействия пользователя. Во время выполнения используется алгоритм 1, если он завершен успешно, то выполняется фиксация значений атрибутов, в противном случае может быть выполнен откат (с помощью журнала отката), и модель вернется в состояние до начала транзакции.

Транслятор выражений построен на основе LR(1) грамматики. Он решает три основные задачи - вычисляет отношения и присваивает новые значения переменным, разбирает выражения на составляющие и строит обратные зависимости между атрибутами, которые образуют граф зависимостей, а также осуществляет преобразование имен атрибутов в уникальные идентификаторы.

Граф зависимостей предназначен для поддержки алгоритма 1. По нему можно установить последовательность конкретных шагов вычисления в рамках алгоритма 1. Также граф используется для удаления объектов из диаграммы, так как в нем хранится информация о принадлежности атрибутов объектам.

Транслятор диаграмм в текст состоит из модуля экспорта семантической сети в дерево данных и модуля обхода этого дерева. Модуль экспорта экспортирует все объекты, атрибуты и значения в XML-документ. Полученный документ содержит в себе полную информацию, по которой можно восстановить диаграмму (но не язык). Модуль обхода построен на основе простого LL(1) транслятора.

В четвертой главе приводится описание прикладных результатов, а именно разработанного редактора ER-диаграмм, и средства визуализации телеметрической информации подсистемы наддува топливных баков ракеты-носителя Союз-2. Важно, что между этими задачами имеются существенные различия, а именно:

- 1) элементы ER-диаграммы обладают сложным визуальным поведением, взаимодействуют с пользователем через диалоговые окна, а визуальное поведение мнемосхемы ограничено изменением значений индикаторов;
- 2) ER-диаграмма конструируется пользователем, мнемосхема всегда имеет одинаковую структуру;
- 3) элементы ER-диаграммы имеют очень простые контуры (линии, прямоугольники), мнемосхема состоит из элементов разнообразной формы (треугольники, закрашенные области, эмблемы и пр.);
- 4) ER-диаграмму необходимо транслировать, в то время как мнемосхема должна интегрироваться с процессами, передающими телеметрическую информацию.

ER-диаграмма состоит из двух основных элементов – сущностей и связей. Рассмотрим для примера сущность – объект диаграммы, представляющий реляционное отношение. Он состоит из описания имен, типов данных и других свойств атрибутов реляционного отношения, а также признаков внешнего ключа.

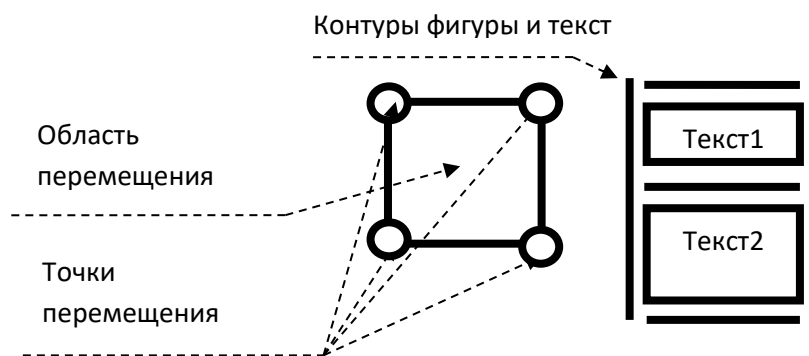


Рис. 3. Визуальная структура объекта “сущность”

Визуальная часть объекта “сущность” строится из элементарных графических форм, представленных на рис. 3.

Таким образом, сущность состоит из пяти управляющих элементов и семи элементов, представляющих собственно внешний вид сущности. В управляющие элементы входят четыре активные точки, с помощью которых осуществляется изменение размера фигуры и активная область, за которую фигура перемещается. Элементы, отображающие информацию, состоят из пяти отрезков прямых, образующих контур фигуры и двух текстовых полей – верхнее представляет название реляционного отношения, а нижнее – таблицу атрибутов.

Визуальная часть объекта “связь” строится из следующих элементарных графических форм (рис. 4). В диссертационной работе показано, что сущность и связь могут быть выражены системой из сорока одного уравнения, представляющих стационарную вычислительную модель. Кроме того, показано, что данная модель может быть преобразована в текст на языке SQL.

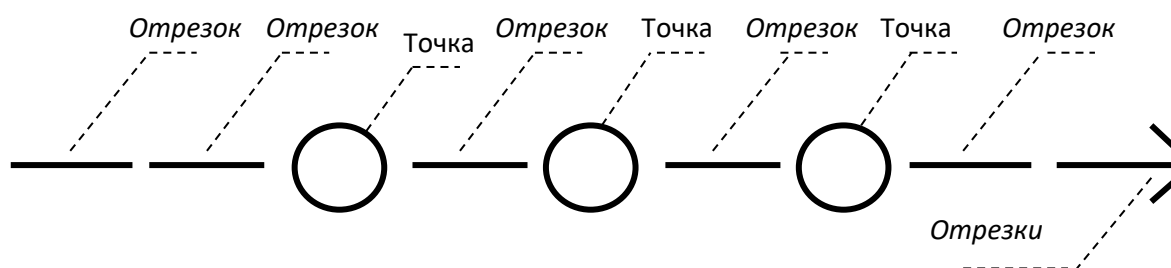


Рис. 4. Визуальная структура объекта “Связь”

При разработке ракетно-космической техники возникают специальные задачи, например, разработка мнемосхем для выполнения контроля ТС различных подсистем. Использование для решения этой задачи вычислительной модели визуального языка продемонстрировано на основе одной из подсистем ракеты-носителя “Союз-2”, а именно мнемосхемы тракта наддува топливных баков. Можно отметить следующие основные отличия задачи от предыдущей:

- 1) тракт наддува баков содержит большее разнообразие элементов, нежели ER-диаграмма; необходимо описать большого количества вариантов поведения при взаимодействии с пользователем для каждого элемента;
- 2) ожидается, что мнемосхема каждый раз при работе имеет одинаковую структуру. При этом представляется нецелесообразным выполнять весь тракт в виде одного

объекта, так как желательно сохранить ряд возможностей, в частности, возможность перемещения элементов.

Для реализации мнемосхемы тракта наддува топливных баков было определено 12 объектов различной сложности. Одним из них является вентиль (рис. 5) для реализации которого требуется двадцать визуальных компонент:

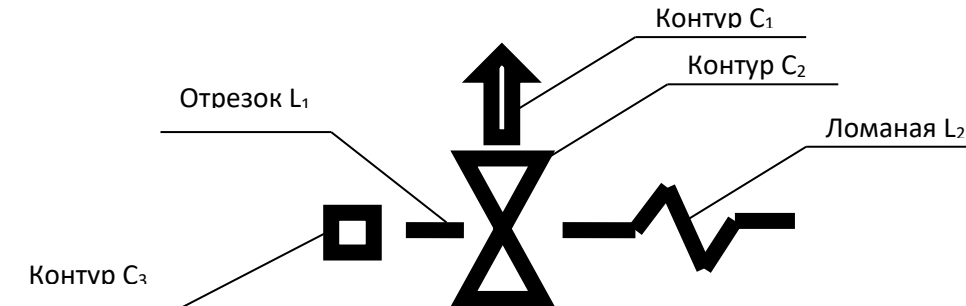


Рис.5. Структура мнемосхемы вентиля.

В качестве средства интеграции источника телеметрии и визуального редактора использована технология сокетов. Редактор открывает серверный сокет, к которому могут присоединиться клиенты, передающие информацию. Каждый клиент передает информацию в виде UNICODE строк определенного формата. Получив такую строку, редактор рассматривает ее как описание воздействия на вычислительную модель и начинает транзакцию, в рамках которой выполняется пересчет модели. Таким образом, СПО, функционирующее на любом компьютере в сети, может телеметрировать параметры в графический редактор, исполняющий мнемосхему, вне зависимости от языка программирования и платформы. При этом управление автоматической оценкой технического состояния производится путем связывания визуальных элементов (цвета индикатора, строковых полей) функцией автоматической оценки корректности параметра.

ЗАКЛЮЧЕНИЕ

В диссертационной работе сформулирована и решена актуальная научно-техническая задача определения универсального графического редактора диаграмм, использующего единую математическую модель для описания предметной области и правил визуализации. Предложенное решение отличается простотой, универсальностью и свободно от использования сложных формализмов, аналогичных формализмам графовых грамматик. Апробирование предложенного подхода произведено путем его применения для построения графического редактора диаграмм, используемых в ходе испытаний и штатной эксплуатации семейства ракет-носителей "Союз-2".

В процессе выполнения исследований в рамках диссертационной работы были получены следующие научные результаты:

- 1) предложена модель визуального языка (вычислительная модель визуального языка) на основе вычислительных моделей Тыугу, позволяющая строить человеко-машинные интерфейсы на основе диаграмм;
- 2) разработан алгоритм расчета модели при воздействии на интерфейс извне, как пользователем, так и внешними приложениями, а также приведения модели к стационарному (детерминированному) виду, при этом возможен автоматический контроль корректности связанных представлений одного СТО;
- 3) разработан метод трансляции диаграмм, заданных вычислительными моделями визуального языка в тексты программ;

- 4) Разработан метод построения интерфейсов между моделью и источником данных, позволяющий, в частности, поддерживать взаимодействие с пользователем и источником телеметрической информации.

При этом можно сделать следующие **выводы и рекомендации**:

А. В области теоретических исследований:

1. Вычислительная модель визуального языка на основе модифицированных вычислительных моделей Тыугу позволяет описать визуальный язык, при этом достигается ряд преимуществ по сравнению с графовыми грамматиками и другими средствами формального описания визуальных языков, а именно: относительная простота решения и единообразие математического аппарата с используемым для описания моделей эталонного функционирования оцениваемых объектов; кроме того, более не требуется выделение абстрактного графа синтаксиса. Модель функционирует в соответствии с предложенным алгоритмом пересчета формализованной семантической сети при взаимодействии с ней. Вычислительную модель всегда можно привести к стационарному виду, то есть к виду, когда вычисления будут приводить к одному и тому же результату независимо от пути.

2. Добавление в вычислительную модель узлов особого вида – графических примитивов – позволяет реализовывать графическое изображение диаграммы. Задача деления диаграммы на объекты также решается добавлением узлов особого вида – логических объектов, связанных иерархическими отношениями. Таким образом, вычислительная модель трактует все свои составляющие единообразно и одновременно может представлять себя тремя способами – смысловым наполнением (данными), графически и в виде взаимодействующих объектов. Показано, что подобная сеть поддерживает интерфейс как с пользователем, так и со сторонними приложениями. При этом диаграмма имеет механизмы для динамического изменения без участия пользователя, причем на всех уровнях представления, а также простое решение для декомпозиции логического объекта в другую диаграмму.

3. Трансляция диаграмм, заданных в вычислительной модели визуального языка, возможна как через преобразование в графовые грамматики, так и через сценарии обхода графа. Таким образом, предложенная модель визуального языка может описывать собственно язык, диаграмму на этом языке и использоваться для трансляции диаграммы в текст, то есть является законченной системой, пригодной для описания редакторов диаграмм.

4. Существенные ограничения модель связаны с отсутствием в ней отношений агрегации и наследования. При этом Алгоритм 2.1 с введением отношения агрегации также требует существенной доработки.

Б. В области прикладных исследований:

1. Доказано (путем моделирования некоторых важных классов визуальных языков), что разработанное программное средство и прикладная вычислительная модель визуального языка позволяют реализовать основные визуальные языки, используемые в отрасли информационных технологий.

2. Вычислительная модель визуального языка пригодна для использования при оценке технического состояния ракетно-космической техники, реализована соответствующая мнемосхема, представляющая одну из подсистем

ракеты-носителя “Союз-2”. При этом она позволяет получать на вход телеметрическую информацию.

3. Библиотека визуальных примитивов требует пересмотра в сторону добавления элементов сложного синтаксиса, так как работа с простыми визуальными примитивами приводит к добавлению в модель большого количества отношений.

4. Целесообразно расширить прикладную вычислительную модель за счет введения высокоуровневых типов данных (множеств, отображений).

5. Еще одним **перспективным направлением дальнейшей разработки темы** является построение интерфейсов визуального редактора как части общей вычислительной модели исполняемого им визуального языка.

6. Техничко-экономическая эффективность при внедрении подхода для анализа технического состояния средств ракетно-космической техники заключается в предоставлении возможности создания мнемосхем инженерам, не имеющим образования в области ИТ, а также в возможности интеграции средств визуализации мнемосхем со сторонними источниками телеметрической информации, потенциально написанными на других языках и находящимся на других компьютерах.

Основной эффект от использования предлагаемого способа заключается в уменьшении сроков и трудоемкости разработки новых средств графического редактирования диаграмм, необходимость в которых возникает в рамках политики импортозамещения, а также при решении узкоспециализированных задач в ракетостроении.

Полученные результаты соответствуют п. 3 «модели, методы, алгоритмы, языки и программные инструменты для организации взаимодействия программ и программных систем» п. 7 «человеко-машинные интерфейсы; модели, методы, алгоритмы и программные средства машинной графики, визуализации, обработки изображений, систем виртуальной реальности, мультимедийного общения» паспорта специальности 05.13.11 «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей».

ОСНОВНЫЕ ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

В рецензируемых журналах из Перечня ВАК и изданиях, приравненных к ним:

1. Степанов П.А. Вычислительная модель визуального языка / П. А. Степанов, М. Ю. Охтилев // Изв. вузов. Приборостроение. 2006. - Т. 49, № 11. - С. 28—32.
2. Степанов П. А. Применение вычислительных моделей для создания редактора диаграмм / П. А. Степанов, М. Ю. Охтилев // Изв. вузов. Приборостроение. 2016. - Т. 59, № 5(28). - С. 939—942. DOI: 10.17586/0021-3454-2016-59-11-939-943.
3. Степанов П.А. Применение вычислительной модели визуального языка к задачам визуального контроля технического состояния ракетно-космической техники / П. А. Степанов //Авиакосмическое приборостроение. 2017. - №5. - С. 28-32.
4. Степанов П.А. Визуализация состояния сложных технических объектов с помощью вычислительных моделей / П. А. Степанов, М. Ю. Охтилев, Б.В.Соколов // Информационно-управляющие системы. 2017. № 6. С. 132–135. doi:10.15217/issn1684-8853.2017.6.132.

5. Степанов П.А. Использование вычислительных моделей для оценки и отображения технического состояния дорожной архитектуры / П. А. Степанов // Автоматизация в промышленности. 2018. - № 4. - С. 29–32.
6. Степанов П.А. Применение вычислительных моделей при обнаружении источников отказов в облачных инфраструктурах / П. А. Степанов // Информатизация и связь. 2019. - № 3. - С. 108–111.

В других изданиях:

1. Степанов П.А. Создание универсального репозитория для хранения диаграмм, используемых CASE – средствами / П.А. Степанов, В.В. Фильчаков // Шестая международная научно-техническая конференция студентов и аспирантов “Радиотехника, электроника и энергетика”: тезисы конференции // МЭИ. - Москва, 2000. - том 1. - С. 302.
2. Степанов П.А. Методы построения универсального репозитория для хранения диаграмм, используемых CASE – средствами / П.А. Степанов // Тезисы докладов 8й международной студенческой школы-семинара “Новые информационные технологии”/ МГИЭМ. – Москва, 2000 - С. 303.
3. Степанов П.А. Подходы к созданию универсального репозитория, используемого CASE – средствами / П.А. Степанов // 3я научная сессия аспирантов ГУАП : тезисы докладов. - ГУАП. - Санкт-Петербург, 2000. - С.161.
4. Степанов П.А. Описание диаграммного языка и разработка генератора диаграмм на основе универсального репозитория / П.А. Степанов, А.В. Бржезовский // Тезисы докладов пятой Санкт-Петербургской Ассамблеи молодых учёных и специалистов. - СПбГУ. – Санкт-Петербург, 2000. - С.45.
5. Степанов П.А. Задание визуального языка с помощью универсального репозитория / П.А. Степанов, А.В. Бржезовский // Труды конференции седьмой Санкт-Петербургской Международной конференции “Региональная информатика -2000”. - СПИИРАН. – Санкт-Петербург, 2001.
6. Степанов П.А. Описание семантики диаграмм на основе отношений между данными графических объектов / П.А. Степанов, А.В. Бржезовский //Седьмая международная научно-техническая конференция студентов и аспирантов “Радиотехника, электроника и энергетика”. - МЭИ. – Москва, 2000. - Том 1. - С. 271.
7. Степанов П. А., Исследование применимости вычислительных моделей для создания редактора диаграмм на примере ER-модели / П.А. Степанов // 19я научная сессия ГУАП. - ГУАП. - Санкт-Петербург, 2016. - С.263-269.
8. Степанов П.А. Эффективная разработка мнемосхем при контроле технического состояния ракетно-космической техники / П.А. Степанов, М.Ю. Охтилев // Труды конференции 19я международная конференция Проблемы управления и моделирования в сложных системах. – ИПУСС РАН. –Самара, 2017 - С.224-229.
9. Степанов П.А. Построение визуальных средств анализа телеметрической информации при оценивании технического состояния космических средств с использованием вычислительных моделей // Труды Восьмой всероссийской научно-практической конференции «Имитационное моделирование. Теория и практика» (ИММОД-2017) (г. Санкт-Петербург, 18-20 октября 2017 г.). - СПб.: Изд-во ВВМ, 2017. - С. 294-298.

Свидетельства о регистрации программ для ЭВМ:

1. Свидетельство о государственной регистрации программы для ЭВМ № 2017615060 “Анализатор и интерпретатор математических соотношений, заданных в вычислительной модели визуального языка”.
2. Свидетельство о государственной регистрации программы для ЭВМ № 2017617893 “Программа построения графа зависимостей переменных для вычислительной модели визуального языка”.
3. Свидетельство о государственной регистрации программы для ЭВМ № 2017617879 “Программа управления внутренней архитектурой вычислительной модели визуального языка”.

Автореферат диссертации

СТЕПАНОВ
Павел Алексеевич

МОДЕЛИ, АЛГОРИТМЫ И ПРОГРАММНЫЕ СРЕДСТВА ОПРЕДЕЛЕНИЯ
ВИЗУАЛЬНЫХ ЯЗЫКОВ НА ОСНОВЕ ВЫЧИСЛИТЕЛЬНЫХ МОДЕЛЕЙ

Текст автореферата размещен на сайтах:

Высшей аттестационной комиссии при Министерстве науки и высшего
образования Российской Федерации
<https://vak.minobrnauki.gov.ru/>

Федерального государственного бюджетного учреждения науки Санкт-
Петербургского института информатики и автоматизации Российской академии
наук (СПИИРАН)
<http://www.spiiras.nw.ru/dissovet/>

Подписано в печать " ____ " _____ 2019 г.
Формат 60x84 1/16. Бумага офсетная. Печать офсетная.
Усл.печ.л. 1,0. Тираж 100 экз.
Заказ №