

# Онтологический подход к организации взаимодействия сервисов интеллектуального пространства при управлении гибридными системами<sup>1</sup>

**Аннотация.** В статье представлен подход к организации взаимодействия сервисов интеллектуального пространства при управлении гибридными системами на примере сценария уборки помещения в системе «умный дом». Технология интеллектуальных пространств используется для поддержки взаимодействия сервисов системы, а для обеспечения интероперабельности при их взаимодействии используется онтологический подход. Для описания поведения каждого отдельно взятого сервиса используется их описание с помощью гибридных автоматов, что позволяет как автоматизировать генерацию программного кода, так и осуществлять верификацию элементов системы.

**Ключевые слова:** гибридные системы, гибридные автоматы, интеллектуальные пространства, онтологии.

## Введение

В последнее время широкое распространение получили прикладные системы (такие как «умный дом», «умный автомобиль», «умный город») [1-3], основывающиеся на технологии интеллектуальных пространств и относящиеся к классу «гибридных систем» (характеризующихся динамикой как дискретной, так и непрерывной природы). Интеллектуальное пространство представляет собой инфраструктуру для взаимодействия сервисов, описывающих физические устройства, которые совместно решают возникающие задачи. Однако данная технология является достаточно новой, поэтому возникла необходимость разработки подхода и математической модели для обеспечения возможности взаимодействия сервисов в интеллектуальном пространстве.

Понятие «гибридная система» является удобным обобщением, предоставляющим понятийный и формальный аппарат для описания и моделирования широкого спектра задач в

различных инженерных приложениях [4, 5]:

- в механических системах: движение объектов, прерываемое их столкновениями;
- в электрических цепях: непрерывные процессы, такие, например, как зарядка конденсаторов, прерываемые включением и выключением тока в цепи;
- в химической промышленности: непрерывный ход химической реакции, управляемый дискретным состоянием клапанов и насосов;
- во встроенных вычислительных системах: цифровой (следовательно, дискретный) компьютер, взаимодействующий с преимущественно аналоговым окружением.

На Рис. 1 приведена общая схема гибридной системы, состояния которой разделены на дискретные и непрерывные составляющие. Изменение состояния системы описывается правилами трёх видов: предикаты состояний (описывают отображение непрерывных состояний в дискретные); логика (описывает правила и условия перехода между дискретными состояниями, в том числе, изменение режима функ-

<sup>1</sup> Представленные результаты исследований являются частью проектов № 13-07-00336, 13-07-12095, 13-07-00271, 12-07-00298, 14-07-00345, 14-07-00363, 14-07-00378, финансируемых РФФИ. Работа также выполнена при государственной финансовой поддержке ведущих университетов Российской Федерации (субсидия 074-U01).

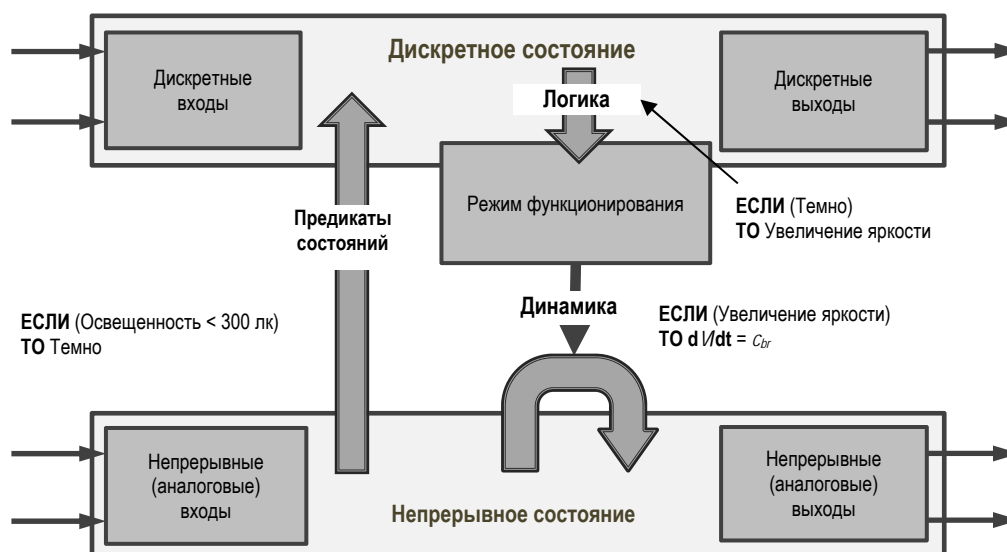


Рис. 1. Структура гибридной системы

ционирования); динамика (описывает законы изменения непрерывных состояний в зависимости от режима функционирования).

Исследования в области формализации и анализа гибридных систем показали, что одним из самых популярных подходов к описанию поведения элементов в таких системах является гибридный автомат. Гибридный автомат – это направленный граф, каждому узлу которого приписана динамическая система или гибридный автомат, а дугам – условия смены поведения и последовательности действий, сопровождающих смену поведения [6]. Описание элементов системы с помощью гибридных автоматов позволяет формализовать их поведение, а также осуществлять верификацию функционирования такого элемента. В то время как использование онтологического описания для взаимодействующих сервисов позволяет осуществлять поддержку их интероперабельности.

В настоящей работе организация взаимодействия сервисов в гибридных системах рассмотрена на примере сценария взаимодействия различных физических устройств умного дома. Каждое из устройств описывается сервисом в интеллектуальном пространстве, что позволяет применить предложенный подход. Пример гибридного автомата, описывающего поведение интеллектуальной системы освещения, позволил формализовать поведение системы освещения в зависимости от тех или иных изменений в интеллектуальном пространстве.

## 1. Современные подходы к формализации и анализу гибридных систем

Широкое распространение и повышенные требования к безопасности систем, обладающих дискретно-непрерывной динамикой (то есть, гибридных), стали причиной активных исследований в области формальных методов описания, анализа и верификации таких систем. Основными направлениями исследований в этой области являются: создание логических и вычислительных моделей и методов, разработка средств компьютерной поддержки формальной спецификации и верификации требований к производительности и безопасности гибридных систем, разработка и синтез корректных (относительно некоторых формальных спецификаций) управляющих программ для гибридных систем [4].

Первым шагом для достижения любой из этих целей является создание формального языка, позволяющего:

- описывать различные типы непрерывных и дискретных процессов, различные способы влияния дискретной динамики на изменение непрерывных переменных, создавать недетерминированные модели и т.д.;
- строить сложные модели объединением простых;
- описывать системы на различных уровнях абстракции, начиная от общих моделей всей

системы в целом, заканчивая их конкретизацией для отдельных составляющих.

На настоящий момент предложено значительное количество методов и средств для формального описания и анализа гибридных систем. В основном, эти средства базируются на автоматном подходе, то есть, описании гибридной системы посредством гибридного автомата [7]. Основным достоинством этого подхода является то, что полученный гибридный автомат может быть использован как для анализа системы с помощью методов имитационного моделирования, так и для анализа с помощью специфических методов теории гибридных автоматов.

Одной из первых систем моделирования гибридных систем на основе гибридных автоматов стала система SHIFT [8]. SHIFT – это язык программирования для описания и имитации динамических сетей гибридных автоматов. То есть систем, состоящих из компонентов, которые могут создаваться, связываться и уничтожаться в процессе работы. Компоненты обладают гибридными свойствами, сочетая фазы непрерывной эволюции со скачкообразными изменениями состояния и могут как функционировать независимо, так и взаимодействовать с другими компонентами посредством общих переменных или событий. Система SHIFT нашла применение в двух основных прикладных областях: системы автоматизированного управления потоком для высокоскоростных автомагистралей и системы управления подводными лодками.

Язык (и система его поддержки) CHARON [9] предназначена для модульной спецификации взаимодействующих гибридных систем. Иерархическое описание структуры системы обеспечивается в CHARON за счет механизмов порождения, сокрытия и параллельной композиции. Для иерархического описания поведения компонент системы CHARON позволяет определять составные состояния посредством механизмов порождения и инкапсуляции. Непрерывное поведение может быть описано дифференциальными и алгебраическими уравнениями и инвариантами, ограничивающими пространство изменения переменных системы. Существует также расширение R-Charon [10], предназначенное для моделирования реконфигурируемых систем. Области применения R-Charon являются модульные реконфигурируемые роботы и большие транспортные системы.

В работе [11] предложена формальная модель гибридных динамических систем Massacio, в которой описание системы складывается из атомарных дискретных компонент (описываемых разностными уравнениями) и атомарных непрерывных компонент (описываемых дифференциальными уравнениями) посредством последовательной и параллельной композиции. Модель Massacio является формальной в том смысле, что она определяет компоненты как математические объекты, доступные для анализа. Структурная композиция компонент может быть использована для структуризации анализа поведения этих компонент.

Подход, предложенный в [12], представляет собой результат конвергенции технологий моделирования гибридных систем и технологий создания предметно-ориентированных языков (Domain Specific Language, DSL). Основная задача, которую пытаются решить авторы [12] – это обеспечение расширяемости языка моделирования. Для решения этой задачи авторы предложили, базовый язык, названный ими Modelyze, и совокупность способов построения прикладных языков моделирования различных классов систем на базе Modelyze. Основными особенностями Modelyze являются определение функций как объектов первого класса, механизм абстрагирования компонентов модели, поддержка символических вычислений при работе с уравнениями. Система типов обеспечивает выявление некоторых ошибок модели путем статического анализа, не прибегая к имитации. Язык Apricot [13] также использует технологию DSL, сочетая в себе особенности предметно-ориентированного и объектно-ориентированного языков.

AnyLogic [14] – популярная коммерческая система имитационного моделирования, обеспечивающая поддержку трёх подходов: дискретно-событийного моделирования, системной динамики и агентного моделирования и допускающая комбинирование концепций из различных подходов в рамках одной модели. Включает в себя графический язык моделирования, а также позволяет расширять созданные модели с помощью модулей, написанных на языке Java. В рамках неё разработан также набор стандартных библиотек для ряда популярных задач моделирования: моделирование бизнес-процессов (производство, цепочки поставок, логистика) и пешеходных потоков.

Другой разработкой, предназначенной для имитационного моделирования гибридных систем, является HyDySS (позднее, EnviCon) [15, 16]. Основной особенностью подхода, лежащего в основе этого пакета, является то, что для описания дискретных (дискретно-событийных) аспектов моделируемой системы используется логический подход, а для описания дискретной динамики – системы продукционных правил относительно логических состояний. Для представления непрерывных процессов используются алгоритмы вычисления транзитивных отношений для элементарных динамических звеньев и основных законов управления. Сложные непрерывные процессы реализуются путем последовательного и параллельного соединения элементарных процессов.

В настоящей работе для формального описания гибридных систем также используется модель гибридного автомата, являющаяся расширением конечного автомата, адаптированным для описания гибридных систем. Описание модели гибридного автомата приведено в системе обозначений, предложенной в [17]. В литературе [18, 7] предложено еще несколько модификаций этого описания.

Гибридный автомат  $H$  – это набор  $H = (Q, X, f, Init, Dom, E, G, R)$ , где:

$Q$  – множество дискретных состояний;

$X \subseteq \mathbb{R}^n$  – множество непрерывных состояний;

$f: Q \times X \rightarrow \mathbb{R}^n$  – векторное поле, описывающее динамику изменения непрерывных переменных автомата;

$Init \subseteq Q \times X$  – множество начальных состояний;

$Dom: Q \rightarrow P^X$  – область допустимых значений непрерывных переменных автомата для каждого из дискретных состояний;

$E \subseteq Q \times Q$  – множество дуг-переходов между дискретными состояниями;

$G: E \rightarrow P^X$  – условия перехода по соответствующей дуге между парой дискретных состояний;

$R: E \times X \rightarrow P^X$  – функция сброса, определяющая изменение значений непрерывных переменных автомата при переходе по соответствующей дуге между парой дискретных состояний.

Гибридный автомат начинает работу в состоянии  $(q_0, x_0) \in Init$ , непрерывное состояние изменяется в соответствии с дифференциальным уравнением:

$$\dot{x} = f(q_0, x),$$

$$x(0) = x_0,$$

а дискретное состояние  $q$  остается постоянным ( $q_0$ ).

Процесс продолжается до тех пор, пока  $x$  принадлежит множеству  $Dom(q_0)$ . Если в какой-то момент времени непрерывное состояние достигает значения  $G(q_0, q_1)$  некоторой дуги  $(q_0, q_1) \in E$ , то дискретное состояние может изменить значение на  $q_1$ . При переходе непрерывное состояние сбрасывается до значения  $R(q_0, q_1, x)$ . После осуществления перехода продолжается изменение непрерывного состояния и так далее.

## 2. Платформа для обеспечения взаимодействия сервисов в интеллектуальном пространстве Smart-M3

Платформа Smart-M3 [19, 20] является одним из вариантов реализации концепции интеллектуальных пространств. Она позволяет создать инфраструктуру, на основе которой распределенные информационно-управляющие системы строят свои интеллектуальные пространства. Smart-M3 объединяет в себе идеи распределенных сетевых систем и Semantic Web. Ее ключевыми идеями являются независимость от конкретных производителей, оборудования, области применения и возможность обмена информацией между различными программными модулями посредством простого и общедоступного информационного брокера. Благодаря использованию Semantic Web в качестве основы, обмен информацией между участниками пространств может осуществляться на основе протокола HTTP и с использованием унифицированных идентификаторов ресурсов (Uniform Resource Identifier — URI) [21].

Общая структура платформы представлена на Рис. 2. Ядро системы подразделяется на 2 элемента: СИБ (семантический информационный брокер, Semantic Information Broker — SIB) и физическое хранилище данных. СИБ предоставляет доступ информационным агентам к информационному пространству, обеспечивая их функциями обработки информации, такими как вставка, извлечение, редактирование, удаление и подписки на изменение информации в

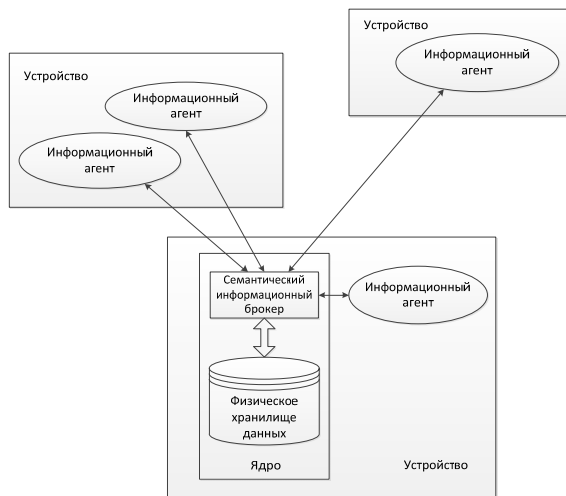


Рис. 2. Структура платформы Smart-M3

интеллектуальном пространстве. В хранилище данных вся информация сохраняется как граф, удовлетворяющий требованиям стандарта RDF (*Resource Description Framework*) [22], описывающего семантические сети, в которых узлы и дуги имеют унифицированные идентификаторы ресурсов. Каждое утверждение, в соответствии с этим стандартом, описывается тройкой «субъект – предикат – объект», например, «Иван – имеет – автомобиль» и является простым предложением. Информационные агенты — это программные модули, с помощью которых осуществляется взаимодействие с семантическим информационным брокером через Smart Space Access Protocol (SSAP — протокол доступа к интеллектуальному пространству) [19, 20].

### 3. Онтологические модели устройств системы «умный дом»

Рассмотрим пример взаимодействия устройств умного дома для сценария уборки помещения. Среди таких устройств в рамках рассматриваемого примера можно выделить (Рис. 3): робот-пылесос, позволяющий строить карту помещения с использованием светочувствительных датчиков и осуществлять уборку с учетом этой карты (например, Yujin Robot iCleo Arte YCR-M05); робот-манипулятор, позволяющий передвигать легкую мебель (столы, стулья) для эффективной уборки помещения (например FESTO Robotino XT), система управления освещением - умный свет, позволяющая изменять интенсивность освещения в помещении в зависимости от текущей ситуации; и система управления сигнализацией, отключающая объемные датчики в убираемом помещении.

Каждое устройство умного дома описывается онтологией, которая формализует основные его возможности и ограничения работы. На Рис. 4 показана онтология робота-пылесоса. В онтологии выделены три класса верхнего уровня: сенсоры робота (Sensors), освещенность помещения (Lightness) и операции, выполняемые роботом (Operation). Освещенность помещения согласно онтологии робота-пылесоса может быть двух типов: достаточная для работы его бесконтактных сенсоров (класс Light) и недостаточная (класс Dark). Сенсоры могут

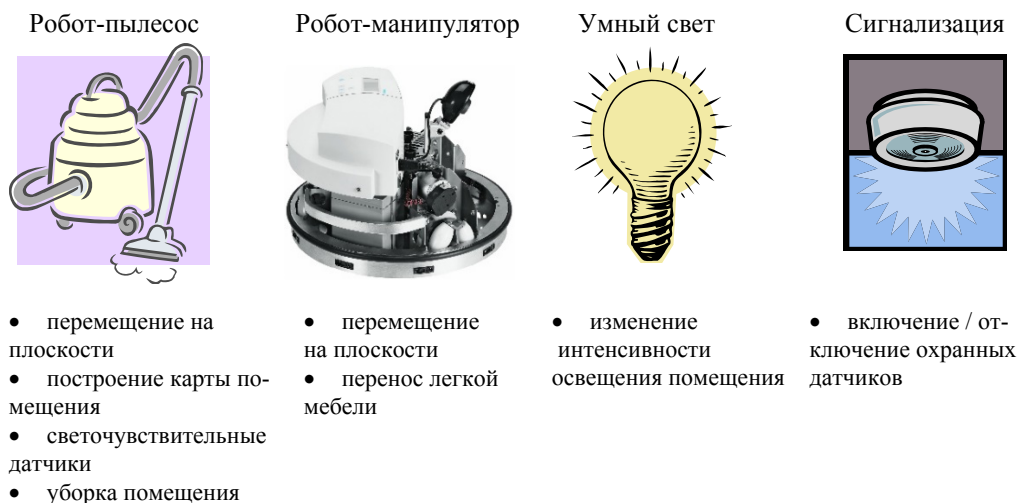


Рис. 3. Основные устройства умного дома для сценария уборки помещения

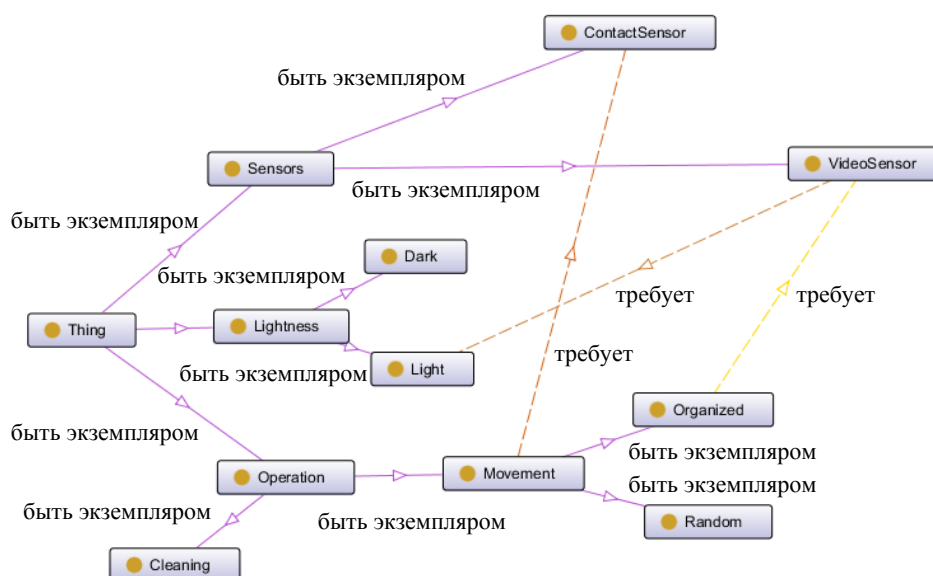


Рис. 4. Онтология робота-пылесоса

быть контактными (класс `ContactSensor`) и бесконтактными (класс `VideoSensor`). Контактные сенсоры функционируют при любой освещенности помещения, в то время как бесконтактные требуют наличия достаточной освещенности в помещении. Операции, выполняемые роботом-пылесосом: всасывание (класс `Cleaning`) и перемещение (класс `Movement`). Причем перемещение может быть как случайным (класс `Random`), с использованием только контактных сенсоров при касании стен или других объектов помещения, так и организованным (класс `Organized`), требующим работы как контактных, так и бесконтактных сенсоров для планирования перемещения робота по помещению.

На Рис. 5 показана онтология робота-манипулятора. В онтологии выделены четыре класса верхнего уровня: физические характеристики робота (класс `Characteristics`), освещенность помещения (класс `Lightness`), сенсоры робота (класс `Sensors`) и операции, выполняемые роботом (класс `Operations`). Физические характеристики робота включают в себя: диаметр (класс `Diameter`), вес робота (класс `TotalWeight`) и грузоподъемность (класс `PayLoad`). Освещенность помещения согласно онтологии робота-пылесоса может быть двух типов: достаточная для работы встроенной камеры (класс `Light`) и недостаточная (класс `Dark`). Сенсоры делятся на три типа: инфракрасный датчик движения (класс

`InfraredDistanceSensor`), оптический сенсор (класс `OpticalSensor`) и встроенная камера (класс `ColourCamera`). Операции, выполняемые роботом-манипулятором включают в себя: перемещение робота по помещению (класс `Movement`) и перенос объектов с места на место (класс `PickAndPlace`).

На Рис. 6 представлена онтология системы умный свет, которая отвечает за управление искусственным освещением в помещении для обеспечения возможности работы светозависимых датчиков работа-манипулятора и робота-пылесоса. Онтология состоит из класса `Sensors`, который включает в себя сенсор освещенности помещения (класс `LightnessLevel`), и класса `Operation`, который включает в себя регулятор искусственного освещения в помещении (класс `LightControl`).

Для эффективной уборки помещения роботу-пылесосу необходима достаточная его освещенность для построения карты и упорядоченного перемещения по помещению. Информация о необходимом уровне освещенности публикуется роботом в интеллектуальном пространстве и становится доступной для системы управления освещением.

Таким образом, если уборка запланирована на темное время суток, система управления освещением автоматически обеспечит требуемый уровень освещенности путем включения искусственных источников света.



Рис. 5. Онтология робота-манипулятора

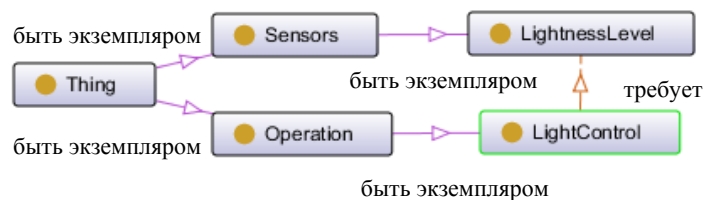


Рис. 6. Онтология системы «умный свет»

При наступлении события «начало уборки помещения» сервис интеллектуального пространства, представляющий робота, переходит в состояние «подготовка к уборке». В этом состоянии он публикует в интеллектуальном пространстве информацию о необходимых условиях для начала уборки помещения. В простейшей модели рассмотрим условие необходимости освещенности помещения не ниже 300 люкс, которое необходимо для работы камеры, по которой робот строит карту помещения.

Сервис интеллектуального пространства, взаимодействующий с физической системой управления освещением, получает уведомления о необходимости контроля уровня освещения и считывает показания с датчика освещенности. В случае если уровня освещения недостаточно (менее чем 300 люкс), сервис переходит в состояние включения освещения.

#### 4. Информационная модель взаимодействия сервисов

Взаимодействие устройств умного дома осуществляется в интеллектуальном простран-

стве на уровне описывающих их сервисов, в то время как сами они функционируют в физическом пространстве умного дома (Рис. 7). Каждое устройство умного дома описывается онтологией, а его поведение гибридным автоматом. Использование онтологий позволяет обеспечить поддержку интероперабельности различных устройств интеллектуального пространства и их синхронного поведения, в то время как описание поведения каждого устройства гибридным автоматом позволяет формализовать такое поведение и предоставляет возможности для верификации поведения устройств интеллектуального пространства.

Рассмотрим ситуацию, при которой в комнате находится пользователь и его мобильное устройство, которое взаимодействует через интеллектуальное пространство с системой управления освещением в помещении для обеспечения, предпочитаемого для него уровня освещения. Мобильное устройство, обладая информацией о том, что пользователь в данный момент времени спит, публикует в интеллектуальном пространстве информацию о комфортном для пользователя на текущий момент уровне освещения.



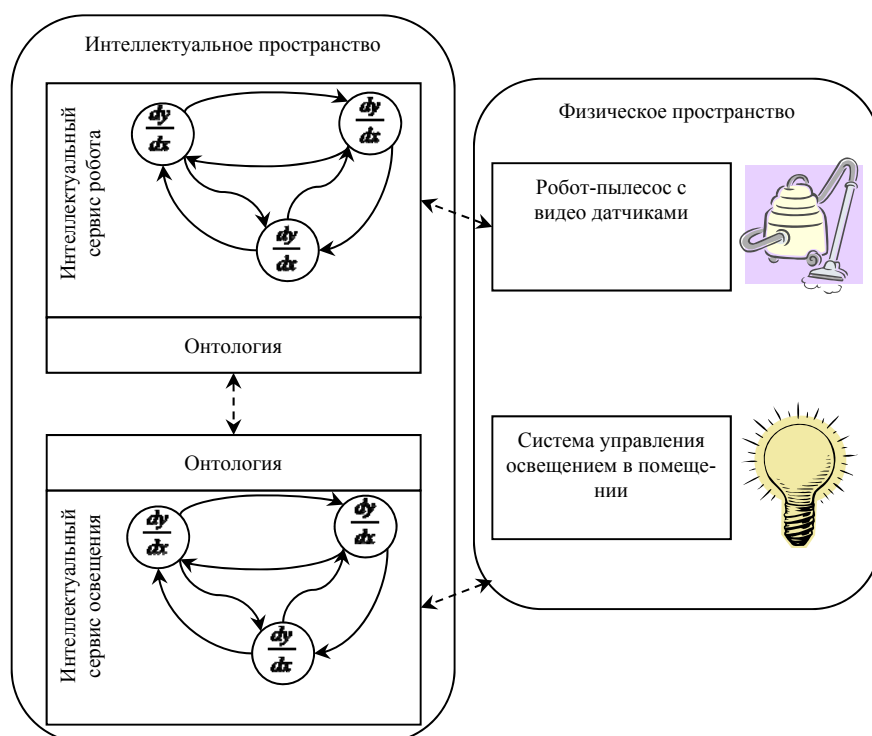


Рис. 7. Информационная модель взаимодействия сервисов на базе интеллектуального пространства с возможностью описания поведения с помощью гибридного автомата на примере робота-пылесоса и системы управления освещением

`([user_id], "http://spiiras.nw.ru#is_a", "user")`

`([user_id],  
"http://spiiras.nw.ru#require_lightness_not_more_than", "10")`

При активации работы некоторого третьего сервиса в интеллектуальном пространстве (например, робота-пылесоса), он публикует свои ограничения в интеллектуальном пространстве (в данном случае необходимость обеспечения требуемого уровня освещения) в терминах своей онтологии.

`([user_id], "http://spiiras.nw.ru#is_a",  
"cleaning_robot")`

`([user_id], "http://spiiras.nw.ru#require_lightness",  
"300")`

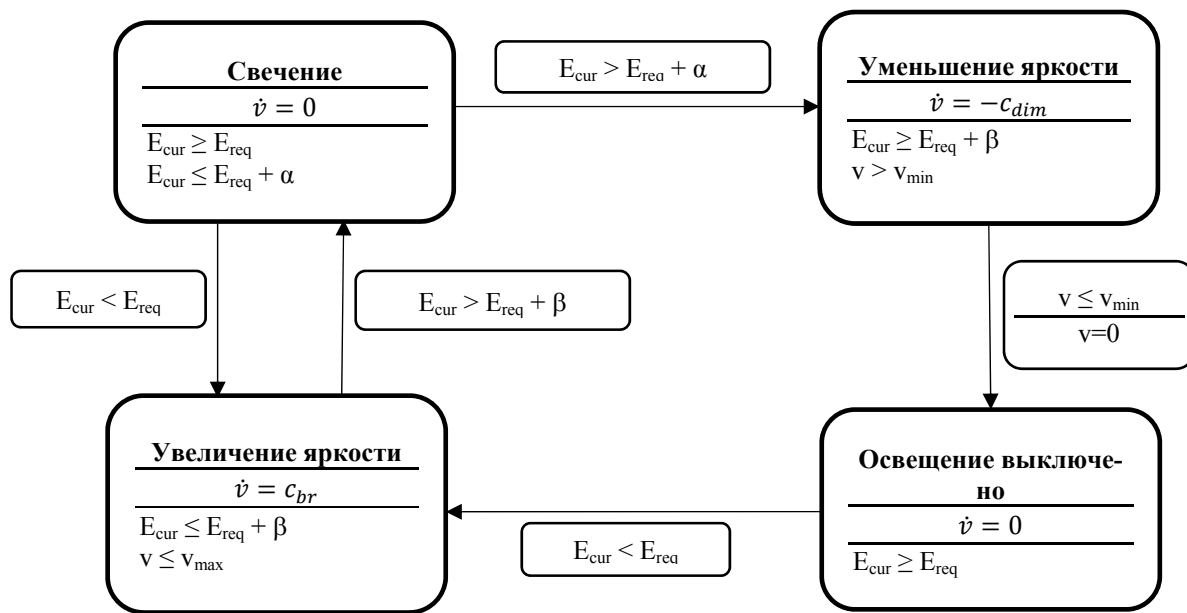
В таком случае система управления освещением должна распознать конфликт между мобильным устройством пользователя и роботом-пылесосом и отдать предпочтение мобильному устройству (оставив освещение помещения на прежнем уровне), а робот-пылесос должен распознать эту ситуацию и запланировать уборку помещения на более позднее время, а в данный момент убрать другое помещение.

На Рис. 8 показан пример гибридного автомата для описания системы управления освещением в помещении. Автомат состоит из четырех состояний: свечение, уменьшение яркости, увеличение яркости, освещение выключено. Каждое из этих состояний описывается дифференциальным уравнением, а переходы между состояниями осуществляются при достижении определенных условий. Например, система управления освещением находится в состоянии «освещение выключено» при текущем уровне освещения ( $E_{cur}$ ) большим, чем требуемый уровень ( $E_{req}$ ). При текущем уровне освещения меньшим, чем требуемый гибридный автомат переходит в состояние «увеличение яркости».

## Заключение

В статье представлен онтологический подход для организации взаимодействия сервисов интеллектуального пространства при управлении гибридными системами на примере сценария уборки помещения в системы «умный дом». Обзор состояния исследований в области формализации и анализа гибридных систем, показал, что





$E_{cur}$  – текущий уровень освещения,  
 $E_{req}$  – требуемый уровень освещения,  
 $\alpha, \beta$  – константы предотвращающие частые переходы между состояниями автомата,  
 $v$  – напряжение на осветительном приборе,  
 $c_{dim}$  – интенсивность уменьшения освещения,

Рис. 8. Пример гибридного автомата, описывающий поведение системы «Умный свет»

в настоящее время для описания поведения различных элементов систем широко используются гибридные автоматы. Они позволяют формализовать поведение таких элементов, а также осуществлять верификацию их функционирования. Использование онтологического моделирования позволяет осуществлять поддержку интероперабельности взаимодействующих в интеллектуальном пространстве сервисов.

Также приведен сценарий взаимодействия различных физических устройств на примере системы «умный дом», а также приведены гибридные автоматы, описывающие устройства системы.

## Литература

1. Гаврилов А.В., Искусственный Домовой, Искусственный интеллект и принятие решений, вып. 2, 2012, с. 77–89.
2. Carreira P., Resendes S., Santos A., Towards automatic conflict detection in home and building automation systems, Pervasive and Mobile Computing, vol. 12, 2014, pp. 37–57.
3. Belley C., Gaboury S., Bouchard B., Bouzouane A., An efficient and inexpensive method for activity recognition within a smart home based on load signatures of appliances, Pervasive and Mobile Computing, vol. 12, 2014, pp. 58–78.
4. Davoren, J.M., Nerode, A. Logics for hybrid systems, Proceedings of the IEEE, vol. 88, issue 7, 2000, pp. 985–1010.
5. Lee E.A., Seshia S.A. Introduction to Embedded Systems, A Cyber-Physical Systems Approach, <http://LeeSeshia.org>, ISBN 978-0-557-70857-4, 2011.
6. Колесов Ю.Б., Сениченков Ю.Б., Инихов Д.Б. Их есть у меня! (Моделирование и исследование сложных динамических систем в MvStudium), Компьютерные инструменты в образовании, №3, С.-Пб.: Изд-во ЦПО "Информатизация образования", 2007, с. 33–39.
7. Henzinger, T.A. The Theory of Hybrid Automata, Eleventh Annual IEEE Symposium on Logic in Computer Science, July 1996, pp. 278–292.
8. Deshpande A., Göllü A., Semenzato L. The SHIFT Programming Language and Run-time System for Dynamic Networks of Hybrid Automata, IEEE Transactions on Automatic Control, vol. 43, issue 4, 1998, pp. 584–587.
9. Alur R., Grosu R., Hur Y., Kumar V., Lee L., Lee I. Modular Specification of Hybrid Systems in CHARON, Proceedings of the 3rd International Workshop on Hybrid Systems: Computation and Control, 2000, pp. 6–19.
10. Kratz F., Sokolsky O., Lee I. R-Charon: a modeling language for reconfigurable hybrid systems, Hybrid Systems: Computation and Control, LNCS 3927, 2006, pp. 392–406.
11. Henzinger T.A. Masaccio: A formal model for embedded components, Proceedings of the First IFIP International Conference on Theoretical Computer Science, LNCS 1872, 2000, pp. 549–563.
12. Broman D., Siek J.G. Modelyze: a Gradually Typed Host Language for Embedding Equation-Based Modeling

- Languages, Technical Report No. UCB/EECS-2012-173, 2012, P.52.
13. Fang H., Zhu H., Shi J. Apricot – An Object-Oriented Modeling Language for Hybrid Systems, <http://arxiv.org/pdf/1304.6498v1.pdf>.
  14. URL: <http://www.anylogic.ru/>
  15. Шпаков В. М. Прототип среды моделирования структурированных совокупностей взаимодействующих процессов // Сборник докладов конференции «Имитационное моделирование. Теория и практика». Санкт-Петербург, 19 – 21 октября 2005. Т. 2. С. 292–295.
  16. Шпаков В.М. Об использовании трансформационных правил для компьютерной реализации непрерывных процессов // Труды СПИИРАН. 2014. Вып. 33. С. 99–116.
  17. Lygeros J., Johansson K.H., Simic S.N., Zhang J., Sastry S.S. Dynamical Properties of Hybrid Automata, IEEE Transactions on Automatic Control, vol. 48, no. 1, January 2003, pp. 2–17.
  18. Lynch N., Segala R., Vaandrager F., Weinberg H.B. Hybrid I/O Automata, In Hybrid Systems III, no. 1066 in LNCS, Springer Verlag, 1996, pp. 496–510.
  19. Smart-M3. Wikipedia. URL: <http://en.wikipedia.org/wiki/Smart-M3>.
  20. Honkela J., Laine H., Brown R., Tyrkkö O. Smart-M3 Information Sharing Platform, 2010 IEEE Symposium on Computers and Communications (ISCC), Italy 2010, pp. 1041–1046.
  21. Berners-Lee T., Fielding R., Masinter L. RFC 3986 – Uniform Resource Identifier (URI): Generic Syntax, URL: <http://tools.ietf.org/html/rfc3986>.
  22. Resource Description Framework (RDF), URL: <http://www.w3.org/RDF>.

**Смирнов Александр Викторович.** Заведующий лабораторией Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН) и заведующий международной лабораторией интеллектуальных технологий для социкиберфизических систем Санкт-Петербургского национального исследовательского университета информационных технологий, механики и оптики (Университет ИТМО). Окончил Ленинградский государственный политехнический университет в 1979 году. Доктор технических наук, профессор, заслуженный деятель науки РФ. Автор более 300 печатных работ. Область научных интересов: управление знаниями, веб-сервисы, системы групповой поддержки принятия решений, виртуальные предприятия, управление цепями поставок. E-mail: [smir@iias.spb.su](mailto:smir@iias.spb.su)

**Кашевник Алексей Михайлович.** Старший научный сотрудник Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН). Окончил Санкт-Петербургский государственный политехнический университет в 2005 году. Кандидат технических наук. Автор 148 печатных работ. Область научных интересов: управление знаниями, профилирование, онтологии, интеллектуальные пространства, логистические системы. E-mail: [alexey@iias.spb.su](mailto:alexey@iias.spb.su)

**Пономарев Андрей Васильевич.** Старший научный сотрудник Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН). Окончил Тюменский государственный нефтегазовый университет в 2003 году. Кандидат технических наук. Автор 22 печатных работ. Область интересов: рекомендующие системы, профилирование, методы дискретной оптимизации, СУБД. E-mail: [ponomarev@iias.spb.su](mailto:ponomarev@iias.spb.su)

**Савосин Сергей Валентинович.** Старший научный сотрудник Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН). Окончил Санкт-Петербургский государственный электротехнический университет в 1995 году. Кандидат технических наук. Автор 11 печатных работ. Область интересов: автоматизированные системы управления, геоинформационные системы. E-mail: [SVSavosin@iias.spb.su](mailto:SVSavosin@iias.spb.su)