

УДК 551.46.077:629.584

Я. С. Пикалёв

Государственное учреждение «Институт проблем искусственного интеллекта», г. Донецк
283048, г. Донецк, ул. Артема, дом 118 б

АНАЛИЗ СУЩЕСТВУЮЩИХ СИМУЛЯТОРОВ РОБОТОТЕХНИЧЕСКИХ СИСТЕМ

Y. S. Pikaliyov

Public institution «Institute of Problems of Artificial intelligence», Donetsk
283048, Donetsk, Artema st., 118 b

ANALYSIS OF EXISTING SIMULATORS FOR ROBOTIC SYSTEMS

Я. С. Пикалёв

Державна установа «Інститут проблем штучного інтелекту», м. Донецьк
283048, м. Донецьк, вул. Артема, буд. 118 б

АНАЛІЗ ІСНУЮЧИХ СИМУЛЯТОРІВ РОБОТОТЕХНІЧНИХ СИСТЕМ

В статье дается определение понятию «симулятор робототехнических систем» (далее – симулятор). Рассматривается структура симуляторов, а также описывается классификация симуляторов. Указываются достоинства и недостатки при работе с симуляторами. Дается обзор наиболее распространенных симуляторов. В ходе обзора дается сравнительная характеристика различных видов симуляторов. Выделяются симуляторы, обладающие наилучшими показателями.

Ключевые слова: робототехника, симулятор, физический движок, графический движок, среда моделирования, виртуальная модель.

The article defines the concept of “robot simulator” (further – simulator). The structure of simulators is considered, and classification of simulators is described. The advantages and disadvantages of working with simulators are specified. An overview of the most common simulators is provided. The review provides comparative characteristics of different types of simulators. Simulators with the best performance are highlighted.

Key words: robotics, simulator, physics engine, graphics engine, simulation environment, virtual model.

У статті дається визначення поняттю «Симулятор робототехнічних систем» (далі — симулятор). Розглядається структура симуляторів, а також описується класифікація симуляторів. Вказуються достоїнства і недоліки при роботі з симуляторами. Дається огляд найбільш поширених симуляторів. В ході огляду дається порівняльна характеристика різних видів симуляторів. Виділяються симулятори, що мають найкращі показники.

Ключові слова: робототехніка, симулятор, фізичний движок, графічний движок, середовище моделювання, віртуальна модель.

Введение

На сегодняшний день стремительно развиваются средства автоматизации различных процессов, всё более широко применяются средства робототехники. Согласно прогнозу Японской ассоциации робототехники, через 5 – 10 лет объем рынка персональной и домашней робототехники достигнет десятков миллиардов долларов [1]. При этом выходу робототехнических систем на потребительский уровень препятствует ряд проблем. Так, рынок робототехники обособлен, производители разрабатывают различные устройства отдельно друг от друга. Нет единого стандарта для применяемых роботизированных систем, вследствие чего робототехнические системы (или их отдельные подсистемы) являются несовместимыми друг с другом, а новые технологии и инструментарий, предназначенный для разработки робототехнических систем, обладает узкой спецификацией. Помимо прочего область робототехники довольно сложна и требует очень высокой квалификации, а обмена опытом и знаниями между различными сферами робототехники практически не происходит [1].

Отдельной проблемой является сложность тестирования и отладки роботизированных систем. Эти процессы требуют значительных затрат времени и ресурсов, особенно при разработке роботов для применения в специальных условиях, например, подводных роботов или роботов для тушения пожаров.

Ускорить и удешевить проектирования робототехнической системы можно с помощью предварительного тестирования алгоритмов управления и конструирования виртуальной модели такой системы. Средства проектирования, имитации и испытания робототехнических систем обобщаются в понятии «симулятор», что частично разрешает вышеописанные проблемы. Симуляторы позволяют решить проблему моделирования особых (опасных или сложно воспроизводимых в реальных испытаниях) условий, а также предоставляют возможность одновременной работы над проектом несколькими членам команды, что невозможно при работе с реальной моделью, так как она обычно выполняется в единичном экземпляре.

Применение симуляторов робототехники в разработке робототехнических систем является перспективным и активно развивающимся направлением в области программных средств поддержки автоматизации и роботизации производства. Данное решение является актуальным и современным, и его используют такие ведущие компании-производители робототехнических систем как: FANUC (Япония), KUKA (Германия), ABB (Швейцария), Kawasaki (Япония), Motoman Robotics (Yaskawa) (США, Япония), OTC Daihen (Япония), Panasonic (Япония), KC Robotics (США); из стран СНГ можно выделить: ООО Промобот, ООО «Сибо Тек», ООО «Подводная робототехника», ООО «РУ.Роботикс», СМП Роботикс, Нэшнл Инструментс Раша Корпорэйшн, ООО «КиберТех Лабс» и другие.

1 Структура и классификация симуляторов

Прежде чем перейти к анализу существующих физических и графических инструментов моделирования элементов компьютерных робототехнических систем, приведем некоторые основные понятия и определения.

Симулятор – имитатор, механический или компьютерный, воссоздающий управление каким-либо процессом, аппаратом или техническим средством. В данном случае – робототехнической системой. Также возможна иная трактовка данного термина: симуляторы – набор программных и аппаратных средств, создающие впечатление действительности, отображая часть реальных явлений и свойств в виртуальной среде.

Симуляция – использование компьютеров для имитации реально существующих ситуаций. Одни симуляторы используются для обучения и тренировки в управлении машинами, другие же являются программами, которые предсказывают (или принимают попытку предсказать) события реального мира [2].

Процесс моделирования (симуляции) сопровождается отображением реального объекта (системы) в модель, изменяющую свое состояние с течением времени. Причём время необратимо, оно не замедляется и не ускоряется. Состояние системы определяется состоянием её элементов, а каждый элемент обладает набором свойств (характеристик).

В имитационном моделировании термин «время» может означать разные понятия:

а) физическое (physical time, T_p) время – это время, которое используется в реальной (физической) системе, которую моделируют.

б) модельное (simulation time, T_s) – это представление физического времени в модели; за единицу модельного времени (h) можно принять временной интервал в 1 минуту, в 10 минут, в 30 минут, в один час и т.д. $T_s = T_p/h$.

в) процессорное (wallclock time, T_w) – время работы симулятора на компьютере.

Одним из преимуществ систем моделирования является возможность задавать скорость выполнения моделирования. Вид моделирования, когда модельное время синхронизируется с процессорным, называется моделированием в реальном времени (real time) [3].

Одним из основных объектов симулятора считается виртуальный робот. Виртуальный робот – это виртуальная модель, имитирующая реальную модель робототехнической системы, воссоздаваемая в виртуальной среде симулятора, где она представлена в виде пространственной модели.

Каждый симулятор включает физический и графический движок. От их возможностей зависит сложность модели роботокомпьютера [4], которую можно реализовать в симуляторе.

Движок (англ. *engine* – мотор, двигатель) – центральная часть компьютерной программы, выполняющая основные функции этой программы. В зависимости от контекста данному понятию может соответствовать различный смысл.

Графический движок – программа, основной задачей которой является визуализация (рендеринг) двухмерной или трехмерной компьютерной графики.

Графический движок работает в режиме реального времени и производит визуализацию с помощью графических процессоров.

Физический движок (англ. *physics engine*) – компьютерная программа, которая производит компьютерное моделирование физических законов реального мира в виртуальном мире, с той или иной степенью аппроксимации. Чаще всего физические движки для физического моделирования используются не как отдельные самостоятельные программные продукты, а как составные компоненты (подпрограммы) других программ.

Физический движок позволяет создать виртуальное пространство, в которое можно добавить виртуальные статические и динамические объекты и указать законы взаимодействия тел и среды. Расчет взаимодействия тел выполняется самим движком. Вычисляя взаимодействие тел между собой и со средой, физический движок приближает физическую модель получаемой системы к реальной и передает уточненные геометрические данные графическому движку.

Симуляторы можно условно разделить на три группы:

- симуляторы без графического движка (текстовые);
- симуляторы, обладающие графическим движком (визуальные);
- симуляторы, обладающие графическим и физическим движком (универсальные).

Есть также специальное программное обеспечение для работы с роботами, которое является лишь дополнением (инструментарием) к симулятору. Это программное обеспечение не оснащено ни физическим, ни графическим движками.

Существующие различные симуляторы обладают как достоинствами, так и недостатками при их использовании для компьютерного моделирования различных систем.

Среди достоинств следует отметить:

- низкую стоимость;
- возможность в любой момент доработать модель;
- возможность отдельно тестировать функциональные составляющие, например, роботов;
- возможность одновременной симуляции нескольких типов роботов и др.

К недостаткам можно отнести:

- в настоящее время даже самый совершенный физический движок не может симулировать все законы реального мира;
- требовательность к ресурсам машины.

2 Обзор современных симуляторов

Рассмотрим наиболее распространенные симуляторы, применяющиеся в робототехнике, возможности их использования для целей управления различными роботами, а также для разработки и конструирования компьютерных роботооператоров.

Следует упомянуть, что многие симуляторы поддерживают лишь определенный вид роботов (определенных производителей или микропроцессоров). Охарактеризуем наиболее известные симуляторы.

Пакет **SolidWorks Simulation** [5] – универсальный инструмент для прочностного анализа конструкций методом конечных элементов. В нём для решения задач симуляции обеспечивается полноценный статический анализ как детали, так и сборки с использованием конечных элементов твердого тела, поверхностей и балок, реализованы разнообразные контактные условия и всевозможные виртуальные соединители. Данная система является отчасти визуальным симулятором. Она в основном используется для проектирования конструкции робота. Поддерживает интеграцию и совместимость со многими популярными симуляторами; поддерживает интеграцию в другие симуляторы путем экспорта данных в файлы с расширением *.SLDASM, *.SLDPR, *.xml, *.stl, *.SLDPRT, *.sdf, *.wrl, *.dae. Отсутствует поддержка языков программирования, то есть отсутствует инструментарий для управления роботом. Является платным продуктом.

NXT-G [6] – является графическим симулятором. Создан специально для популярного робототехнического комплекта Lego Mindstorms NXT. Используется для программирования модуля NXT Brick. Данный симулятор имеет интуитивно понятный интерфейс, управление роботами осуществляется с помощью специальных блоков, размещаемых на LEGO-балках вдоль оси последовательности действий. Порядок выполнения программы определяется порядком следования блоков. Является бесплатным продуктом.

TrikStudio [7] – графический симулятор, ориентированный на программирование учебных роботов фирмы Trik. Включает в себя симулятор, который позволяет тестировать программы без использования реального робототехнического набора. Отличительной особенностью TRIK Studio является интерактивный режим имитационного моделирования. Является бесплатным продуктом.

Самым крупным и известным симулятором является комбинация пакета **Simulink** и **Robotics System Toolbox** [8], входящие в состав программного продукта **MATLAB**. Данный продукт обеспечивает алгоритмы и аппаратные подключения для разработки автономных мобильных робототехнических приложений. Этот пакет обладает обширной библиотекой разнообразных технических средств и устройств, большинство из которых доступно для испытания на персональном компьютере. Алгоритмы Toolbox включают в себя представление карты движения, планирование маршрута, и последующего пути для мобильных роботов. Также можно проектировать и создавать прототипы управления двигателем, реализовывать компьютерное зрение для робота. Поддерживает как интеграцию сторонних проектов симуляторов и CAD-систем, а именно форматов *.xml, *.stl, а также различные GUI-интерфейсы и пользовательские драйвера (CMex-функции); поддерживает язык программирования C/C++. Относится к категории универсальных симуляторов. Является платным продуктом.

AnyCode Marilou Robotics Studio (далее – Marilou) [9] – среда разработки и симулирования мобильных роботов и манипуляторов с учетом физических законов реального мира. Поддерживает контроль и управление над следующими физическими параметрами: масса, упругость, размеры материала, вращающий момент и т.п. Marilou позволяет подключать к роботу различные виртуальные устройства: компас, акселерометры, двигатели и сервомоторы, бампер, сенсоры расстояния (ультразвуковой и инфракрасный), GPS и другие устройства.

В редакторе объектов Marilou доступны статические и динамические объекты, которые можно размещать в симулируемом мире (поддерживается одновременная симуляция нескольких роботов). Сложные объекты в Marilou строятся из более простых (используется иерархический подход к представлению объекта), что позволяет повторно использовать части объектов.

Основой Marilou является технология MODA (Marilou Open Devices Access) — SDK для работы с роботами и их компонентами в симуляторе. После синхронизации с симулятором алгоритмы управления роботом могут запускаться на компьютере. Программирование алгоритмов управления роботом возможно с помощью языков C/C++, C++ CLI, C#, J#, VB#. Является условно-бесплатным продуктом.

Arduino IDE [10] является текстовым симулятором, который позволяет составлять программы в удобном текстовом редакторе, компилировать их в машинный код, и загружать на различные версии микропроцессоров Arduino. Является бесплатным и кроссплатформенным продуктом, поддерживает популярные языки программирования: C/C++, C#, Java, Python.

OROCOS (Open Robot Control Software project) [11] – текстовый симулятор. Является бесплатным продуктом и используется в качестве дополнительного инструментария для различных симуляторов. Поддерживает язык программирования C/C++.

ROS (Robot Operating System) [12] – является универсальным симулятором с открытым исходным кодом (лицензия BSD – т.е. возможно использовать и модифицировать программу под свои задачи, в т.ч. в коммерческих целях).

Главная возможность ROS – это повторное использования кода в робототехнических исследованиях и разработках. Данный продукт поддерживает интеграцию различных драйверов, алгоритмов и популярных открытых робототехнических библиотек. Помимо этого ROS обладает большим функционалом: аппаратная абстракция, низкоуровневый контроль оборудования, реализация часто используемого функционала, передача сообщений между процессами, управление пакетами. Однако

ROS не является системой реального времени, хотя и может использовать системы реального времени (например, OROCOS Real-time Toolkit). ROS является распределённой системой процессов. Также данный продукт легко интегрируется с различными симуляторами (на данный момент интегрирован с OpenRAVE, OROCOS и Player). Поддерживает языки программирования C++ и Python (также имеются тестовые библиотеки на LISP, Octave Java, Lua). Также ROS обладает встроенным пакетом для тестирования – rostest, что облегчает тестирование приложений. В настоящее время ROS работает только под UNIX-подобными системами.

V-REP [13] является одним из лучших решений среди симуляторов, обладает большим количеством функций, а также сложным API. Универсальный симулятор робототехники V-REP, с интегрированной средой разработки, основан на распределённой архитектуре управления: каждый объект/модель может управляться индивидуально с помощью встроенного сценария, плагина, узла ROS, клиента удалённого API или пользовательского решения. Подходит для применения в нескольких роботах. Программный код может быть написан на C / C ++, Python, Java, Lua, Matlab, Octave, но генерацию кода нужно реализовать через API, что является трудоёмким процессом. Используется для быстрой разработки алгоритмов, моделирования автоматизации производства, быстрого прототипирования и верификации робототехники, удалённого мониторинга и т.д. Поддерживает интегрирование сторонних симуляторов, а также различные GUI-интерфейсы и пользовательские драйвера.

RobotC [14] – текстовый симулятор, является лидером среди языков программирования для изучения роботов и подготовки к соревнованиям. Он основан на языке программирования C, и обладает простой в использовании средой разработки. Является платным программным обеспечением.

BricxCC [15] – текстовый симулятор, наиболее распространённый инструмент, поддерживающий язык программирования NXC. Это свободно распространяемая программа, имеющая большое число различных инструментов для работы с блоками Lego Mindstorms, фактически может полностью заменить стандартное программное обеспечение Lego (кроме драйверов). Встроенные библиотеки языка позволяют работать с устройством на различных уровнях, присутствуют низкоуровневые средства обращения к входам и выходам устройства, обращение к физическим адресам памяти, а также высокоуровневые команды управления моторами и получения данных с датчиков.

Robotino View [16] – универсальный симулятор, созданный Festo Didactic для обучения робототехнике. Программировать робота можно при помощи современных языков программирования: C/C++, C#, Java, а также имеется возможность интеграции симулятора с MATLAB, ROS, SmartSoft, MRDS. Является бесплатным продуктом.

Microsoft Robotics [17] – это пакет программ, который может использоваться для управления различными роботами и включает в себя полноценный универсальный симулятор. Поддерживает языки программирования VPL, C#. Данный симулятор позволяет работать со следующими виртуальными устройствами: GPS, лазерный дальномер, инфракрасный дальномер, компас, сенсор цвета, сенсор яркости, веб-камера. Является условно-бесплатным продуктом. Компоненты в Robotics Studio представлены в виде независимо исполняемых сервисов в виде сервисов с GUI-интерфейсом. Протокол SOAP, использующийся для взаимодействия распределённых сервисов, не предназначен для приложений, работающих в режиме реального времени. Поддерживает импорт следующих типов CAD-файлов: *.dae, *.obj и *.x. Помимо этого поддерживает интеграцию в различные симуляторы через GUI- и COM-интерфейсы и пользовательские драйвера.

Gazebo [18] – универсальный симулятор, разработанный для операционной системы Linux. Является бесплатным продуктом. Данное решение поддерживает симуляцию нескольких роботов с сенсорами в окружении различных объектов, а также поддерживает следующие моделируемые сенсоры: лазерный дальномер, камера, устройство для чтения RFID-меток. Поддерживается большинство робототехнических систем. В Gazebo встроена возможность чтения файлов в формате Collada, что позволяет добавлять в симулятор объекты, спроектированные в одном из редакторов 3D-моделей. Поддерживает языки программирования C/C++, Python. Поддерживает как экспорт собственных проектов в сторонние симуляторы, так и импорт сторонних симуляторов посредством форматов *.xml, *.stl, а также через GUI-интерфейсы и пользовательские драйверы.

Algodoo [19] – универсальный 2D-симулятор. В данном продукте можно моделировать различные физические объекты, пружины, оптические устройства и т.п. В Algodoo встроены скриптовый язык программирования Thyme. Является бесплатным продуктом.

URBI [20] – это текстовый симулятор. Является кроссплатформенным решением, поддерживает язык программирования C/C++. URBI основывается на распределенной компонентной архитектуре UObject. Она также включает urbiScript – параллельный и событийный скриптовый язык.

В основе URBI лежит метод, отделяющий управляющую программу на urbiScript от взаимодействия с ОС и т.д. с помощью прослойки из UObject-драйверов. Т.е. для контроля робота при помощи URBI нужно создать промежуточные UObject-драйверы для своего оборудования. Обладает интеграцией с различными симуляторами.

USARSim (Unified System for Automation and Robotics Simulation) [21] является универсальным симулятором и бесплатным решением, основан на технологии Unreal Engine 2.0, но без приобретенной копии Unreal Engine, которая – платная, не функционирует. Контроль над роботами, которые смоделированы в USARSim, может быть осуществлен на любом языке программирования, который поддерживает TCP сокет. Является кроссплатформенным решением. Поддерживаются различные геометрические модели и объекты для создания модели робота. Помимо этого поддерживает моделирование различных интерфейсов (GPS и т.п.). USARSim применяется при моделировании роботов, использующихся для сбора данных с Марса и Луны.

Webots [22] является универсальным симулятором. Данный продукт использует физический движок Open Dynamics Engine (ODE). Является платным продуктом. Предусматривает моделирование нескольких роботов в одной сцене запуска. Поддерживает языки программирования C/C++, Java, Python. Webots позволяет моделировать робота посредством различных геометрических моделей и объектов, а также посредством различных интерфейсов и датчиков. Обладает библиотекой созданных моделей различных роботов.

robotSim [23] является универсальным симулятором, который позволяет моделировать робота, используя различные объекты и датчики. Помимо этого это решение предусматривает импорт CAD-файлов (COLLADA или 3DS). Управление моделью робота обеспечивается C++ API, LabVIEW, или другими языками программирования, поддерживающие работу с сокетами. Является платным решением.

SARGE (Search and Rescue Game Engine) [24] является универсальным симулятором. Он был разработан для обучения правоохранительных органов в использовании робототехники в поисково-спасательных работах. Обладает лицензией Apache License V2.0. В качестве графического движка используется Unity, а в качестве физического движка – PhysX. SARGE поддерживается операционными системами Windows

и Mac. Помимо всего прочего при моделировании робота имеется возможность работы с такими устройствами, как 3D-камера, компас, GPS и т.п. Но, к примеру, для применения GPS-систем необходима поддержка Google Earth.

Ipzrobots [25] является универсальным симулятором, который находится под GPL лицензией (бесплатный продукт с открытым исходным кодом), а также распространяется для платформ Linux и Mac. Данный продукт использует ODE и OSG (OpenScreenGraph) движки.

SimRobot [26] является бесплатным универсальным симулятором. Использует ODE и OpenGL движки. Обладает библиотекой различных объектов робототехники, но помимо этого обладает инструментарием для создания пользовательских роботов, а также содержит ограниченный набор различных устройств и датчиков: камера, датчик расстояния, датчик давления и датчик, определяющий значения углов поворота моторов.

SubSim [27] является текстовым симулятором, применяется при использовании контроллеров EyeBot в автономных подводных аппаратах. Использует физический движок Physics Abstraction Layer (PAL). Модели различных аппаратов могут быть импортированы из CAD-файлов Milkshape3D. Для программирования используется язык C/C++.

EyeSim [28] является двумерным текстовым симулятором, применяющийся для роботических платформ EyeBot. Для реализации функций симулятора использует библиотеку RoBIOS (Robot BIOS). Для создания модели робота и окружающей среды использует текстовые файлы Wall, использующий четыре значения для представления начальной и конечной позиции в двумерной системе координат), или Maze, в котором буквально рисуется карта движения в текстовом файле при помощи символов подчеркивания, а также другие символы.

Сравним характеристики симуляторов, отдельно рассматривая симуляторы различных типов: текстовые, графические, универсальные.

Характеристики текстовых симуляторов приведены в табл. 1. Все они характеризуются низкой ресурсоёмкостью, относительно простым и интуитивно понятным интерфейсом. Большинство из них использует в качестве аппарата управления языки программирования C/C++, практически отсутствует возможность экспорта/импорта проектов различных симуляторов; узконаправленность.

Ключевым свойством графических симуляторов является их интерактивность; помимо этого, можно выделить следующие характеристики: как и текстовым симуляторам, графическим свойственна узконаправленность; отсутствует поддержка языков программирования.

Универсальным симуляторам свойственна высокая и средняя ресурсоёмкость; возможность работы с любым типом робота; наличие библиотек с различными эталонами роботизированных механизмов; поддержка таких популярных языков программирования, как C/C++, C#, Python, Java; наличие инструментария для тестирования моделей роботов и различных интерфейсов. Их характеристики приведены в табл. 2 и 3. Они разделены по критерию «ресурсоёмкость», которая определяется уровнем требований к видеокarte.

К группе симуляторов со средней ресурсоёмкостью отнесены системы с требованиями: 64 Mb VRAM, 1024 Mb RAM, 2.2 GHz. К симуляторам с высокой ресурсоёмкостью отнесены симуляторы с требованиями: 256 Mb VRAM, 1024 Mb RAM, 2.2 GHz.

Исходя из проведенного анализа можно выделить следующие актуальные симуляторы: AnyKode Marilou Robotics Studio, V-REP, Gazebo.

Таблица 1 – Сравнительная характеристика текстовых симуляторов

Название среды	Экспорт/ импорт	Достоинства	Недостатки	Область применения
RobotC	–	Он основан на языке программирования C(CI) и обладает простой в использовании средой разработки	Является платным решением, обладает недостаточным инструментарием	Данный продукт предназначен для академической, любительской или коммерческой разработки.
VicxCC	–	Удобный интерфейс. Комбинирование низкоуровневых и высокоуровневых команд. Точность управления	Ошибки во внутреннем коде программы, обладает недостаточным инструментарием	Данный продукт предназначен для академической, любительской или коммерческой разработки.
URBI	–	Поддержка языка программирования C/C++, кроссплатформенность	Отсутствие собственного симулятора	Данный продукт предназначен для академической, любительской или коммерческой разработки
OROCOS	–	Является open-source проектом, поддерживает язык программирования C/C++	Является набором библиотек, отсутствует инструментарий для конструирования и симуляции	Данный продукт предназначен для академической, любительской или коммерческой разработки.
Arduino IDE	–	Поддержка популярных языков программирования. Точность управления микроконтроллером	Ориентирован на платформу Arduino.	Программирование роботов на базе Arduino.
SubSim	CAD-файлы Milkshap e3D	Используется специальный физический движок; поддерживает язык программирования C/C++	Ориентирован на платформу EyeBot	Применяется при использовании контроллеров EyeBot в автономных подводных аппаратах
EyeSim	CAD-файлы Milkshap	Поддержка специальной библиотеки для реализации функционала модели робота	Является двухмерным симулятором; узконаправленность	Применяется для роботических платформ EyeBot

Таблица 2 – Сравнительная характеристика универсальных симуляторов средней ресурсоёмкости

Название среды	Экспорт/импорт	Достоинства	Недостатки	Область применения
Przrobots	–	Наличие специального графического и физического движков, бесплатное решение	Нет информации об инструментарии для конструирования модели робота, а также интерфейсов.	Предназначен для академической, любительской или коммерческой разработки.
ROS	–	Поддерживает популярные языки программирования, решение с открытым исходным кодом, имеет собственный пакет для тестирования	Совместим только с Ubuntu Linux, не является real-time системой	Предназначен для академической, любительской или коммерческой разработки.
Algodoo	–	Поддерживает язык программирования Thyme, который добавляет большую свободу действий в симуляторе, обладает отличной визуализацией и физикой	Является 2D-симулятором	Менее распространён из-за отсутствия 3D-симуляции
Robotino	–	Поддерживает популярные языки программирования,	Поддерживает лишь работа от фирмы Festo Dynamic, обладает ограниченным инструментарием	Предназначен для обучения робототехнике
Webots	*. Wrl	Обладает инструментарием для тестирования моделей роботов и различных интерфейсов; поддерживает программное управление посредством популярных языков программирования.	Является платным решением, обладает ограниченным функционалом инструментария для конструирования модели робота	Предназначен для академической, любительской или коммерческой разработки.
robotSim	*.ode, *.3ds	Поддержка C++ API; обладает инструментарием для тестирования моделей роботов и различных интерфейсов.	Платное решение; обладает ограниченным функционалом инструментария для конструирования модели робота	Предназначен для академической, любительской или коммерческой разработки.
AnyCode Marilou Robotics Studio	Импорт: *.obj, *.stl, *.x Экспорт: *.mphx, *.world.	Множество возможностей для разработки и модификации роботов, поддержка языков программирования C/C++, C#, и различных интерфейсов. Простой, интуитивно понятный интерфейс.	В home-версии существует ограничение по объектам (120), отсутствие поддержки инструментария инверсной кинематики.	Предназначен для академической, любительской или коммерческой разработки.

Таблица 3 – Сравнительная характеристика универсальных симуляторов высокой ресурсоёмкости

Название среды	Экспорт/ импорт	Достоинства	Недостатки	Область применения
SARGE	–	Обладает инструментарием для тестирования моделей роботов и различных интерфейсов; обладает специальным графическим движком; бесплатное решение.	Необходимость дополнительного ПО для некоторых типов устройств; обладает ограниченным функционалом инструментария для	Обучение правоохранительных органов в использовании робототехники в поисково-спасательных работах
SimRobot	–		Обладает ограниченным функционалом инструментария для конструирования модели робота.	
USARSim	–	Специальный графический движок; совместимый с языками программирования, поддерживаемыми TCP-сокеты; кроссплатформенность; инструментарий для тестирования моделей роботов и различных интерфейсов	Без приобретенной копии Unreal Engine, является платным продуктом; обладает ограниченным функционалом инструментария для конструирования модели робота.	Предназначен для академической, любительской или коммерческой разработки.
Microsoft Robotics	Импорт: *.dae, *.obj, *.x. Экспорт: *.dae	Огромное количество возможностей для разработки и модификации робототехнических систем, поддержка различных интерфейсов, в т.ч. GUI-интерфейсов популярных языков программирования, а также различных интерфейсов.	Сложность определения переменных; громоздкость программ; невозможна симуляция в режиме реального времени.	
V-REP	Импорт: *.dxf, *.3ds, *.obj, *.stl, *.dae, *.urdf Экспорт: *.itt, *.ftb, *.dae, *.ik, *.obj, *.dxv, *.stl	Огромное количество возможностей для разработки и модификации робототехнических систем, поддержка популярных языков программирования, а также различных интерфейсов.	Использует скриптовый язык Lua, подключение других языков программирования проблематично. Проблема с изменением положения объекта по оси Z.	
Gazebo	*.xml, *.stl.	Бесплатный продукт, обладает симулятором и инструментарием для создания конструкций	Совместим только с Ubuntu Linux	
Simulink и Robotics System Toolbox	*.xml, *.stl, *.SLDPR, *.SLDASM, *.SLDPRT	Поддержка популярных языков программирования, имеет инструментарий совместимый с популярными симуляторами	Платное решение, сложный интерфейс, необходимо наличие знаний в области робототехники	

Выводы

Использование инструментария симуляторов обладает рядом преимуществ: симуляторы обладают графическим или физическим движком, или же их комбинацией, которые являются важным инструментарием для осуществления симуляции, что предоставляет разработчикам возможность создавать или модифицировать произведенные робототехнические системы без отрыва от их производства, позволяет реализовывать и оптимизировать различные идеи до проведения тестов на реальной робототехнической системе с целью их внедрения; предоставляет инструментарий для анализа различных проектов (сценариев), а также их последовательностей; обладают интуитивно понятной визуальной средой для демонстрации продукта, отображение реализации идеи разработки робототехнической системы в различных средах и при использовании в различных процессах.

Главным недостатком симуляторов является проблема с их соответствием реальным моделируемым робототехническим системам, а также окружающим средам и различным процессам. То есть это значит, что разработчик должен провести точную настройку всех параметров в симуляторе на основании реальной информации. Если вышеуказанные действия будут выполнены, а также в ходе симуляции виртуальный робот при взаимодействии с моделируемыми средами и процессами покажет положительный результат, то код такой программы может быть внедрен в реальном прототипе с минимальными изменениями.

Список литературы

1. Microsoft Robotics Studio – робототехника для всех [Электронный ресурс] / BYTEMag.ru – URL: <https://www.bytemag.ru/articles/detail.php?ID=6470> – Дата обращения 15.11.2016.
2. Gibilisco, S. Concise encyclopedia of robotics [Текст] / S. Gibilisco // The McGrawHill Companies, Inc. 2003
3. Лекция 7 : Распределенное имитационное моделирование [Электронный ресурс] / Национальный Открытый Университет "ИНТУИТ". URL: http://www.intuit.ru/studies/courses/1146/238/lecture/6150?page=1&keyword_content=discrete%20event%20simulation – Дата обращения .09.2016.
4. Иванова, С. Б. Роботизированные компьютерно-аппаратные комплексы широкого назначения: необходимость и проблемы создания [Текст] / С. Б. Иванова, И. С. Сальников, Р. И. Сальников // Искусственный интеллект. – 2015. – № 0 (1). – С.50–62.
5. Robotics [Электронный ресурс] / 3D CAD Design Software | SOLIDWORKS. URL: <http://www.solidworks.com/sw/education/robotic-design-resources-students.htm> – Дата обращения 10.10.2016.
6. NXT-G: the development environment supplied with Lego Mindstorms, NXT-G [Электронный ресурс] / Génération Robots. URL:<http://www.generationrobots.com/blog/en/2015/09/nxt-g-the-development-environment-supplied-with-lego-mindstorms-nxt-g> – Дата обращения 10.10.2016.
7. TRIK Studio [Электронный ресурс] / Всё о ТРИК: TRIK Studio. URL: <http://blog.trikset.com/p/trik-studio.html> – Дата обращения 10.10.2016.
8. Robotics System Toolbox [Электронный ресурс] / MathWorks - Makers of MATLAB and Simulink. MATLAB & Simulink. URL: <https://www.mathworks.com/products/robotics> – Дата обращения 10.10.2016.
9. anyKode Marilou [Электронный ресурс] / anyKode Marilou - Modeling and simulation environment for Robotics. URL: <http://doc.anykode.com/frames.html?frmname=topic&frmfile=index.html> – Дата обращения 21.11.2016.
10. What is Arduino? [Электронный ресурс] / Arduino. URL: <https://www.arduino.cc/en/Guide/Introduction> – Дата обращения 10.10.2016.

11. The OROCOS Project [Электронный ресурс] / Departement Werktuigkunde - KU Leuven. URL: <https://people.mech.kuleuven.be/~orocos/pub/documentation/rtt/v1.12.x/doc-xml/orocos-overview.html> – Дата обращения 10.10.2016.
12. Core Components [Электронный ресурс] / ROS.org | Powering the world's robots. URL: <http://www.ros.org/core-components> – Дата обращения 10.10.2016.
13. V-REP User Manual [Электронный ресурс] / Coppelia Robotics V-REP: Create. Compose. Simulate. Any Robot. URL: <http://www.coppeliarobotics.com/helpFiles/index.html> – Дата обращения 01.11.2016.
14. ROBOTC Downloads [Электронный ресурс] / ROBOTC a C Programming Language for Robotics. URL: <http://www.robotc.net/download> – Дата обращения 10.10.2016.
15. Bricx Command Center 3.3 [Электронный ресурс] / SourceForge - Download, Develop and Publish Free Open Source Applications. URL: <http://bricxcc.sourceforge.net> – Дата обращения 10.10.2016.
16. Robotino® View [Электронный ресурс] / Festo Didactic. URL: <http://www.festo-didactic.com/int-en/services/robotino/programming/robotino-view> – Дата обращения 10.10.2016.
17. Welcome to Robotics Developer Studio [Электронный ресурс] / MSDN: Learn to Develop with Microsoft Developer Network. URL: <https://msdn.microsoft.com/en-us/library/bb648760.aspx> – Дата обращения 10.10.2016.
18. Beginner: Overview [Электронный ресурс] / Gazebo. URL: http://gazebosim.org/tutorials?cat=guided_b&tut=guided_b1 – Дата обращения 10.10.2016.
19. What is it? [Электронный ресурс] / Algodoo. URL: <http://www.algodoo.com/what-is-it> – Дата обращения 10.10.2016.
20. Urbi Open Source [Электронный ресурс] / Gostai. URL: <http://www.gostai.com/products/urbi> – Дата обращения 10.10.2016.
21. System Overview [Электронный ресурс] / SourceForge - Download, Develop and Publish Free Open Source Applications. URL: http://usarsim.sourceforge.net/wiki/index.php/3._System_Overview – Дата обращения 10.10.2016.
22. Webots [Электронный ресурс] / Cyberbotics. URL: <https://www.cyberbotics.com/overview> – Дата обращения 10.10.2016.
23. robotSim 3D Simulator for NI Robotics Starter Kit - Cogmation Robotics [Электронный ресурс] / National Instruments: Test, Measurement, and Embedded Systems. URL: <http://sine.ni.com/nips/cds/view/p/lang/ru/nid/209028> – Дата обращения 10.10.2016.
24. About SARGE [Электронный ресурс] / SourceForge - Download, Develop and Publish Free Open Source Applications. URL: <http://sarge.sourceforge.net/page11/page11.html> – Дата обращения 10.10.2016.
25. Documentation of Iprrobots [Электронный ресурс] / Research Network for Self-Organization of Robot Behavior. URL: <http://robot.informatik.uni-leipzig.de/software/lprrobots-0.4-doc/html/main.html> – Дата обращения 10.10.2016.
26. SimRobot – A General Physical Robot Simulator and its Application in RoboCup [Электронный ресурс] / Research Network for Self-Organization of Robot Behavior. URL: <http://www.informatik.uni-bremen.de/kogrob/papers/rc06-sim.pdf> – Дата обращения 10.10.2016).
27. SubSim Overview [Электронный ресурс] / Robotics UWA. URL: <http://robotics.ee.uwa.edu.au/auv/subsim/doc/About.html> – Дата обращения 10.10.2016.
28. EyeSim - EyeBot Simulator [Электронный ресурс] / Robotics UWA. URL: <http://robotics.ee.uwa.edu.au/eyebot/doc/sim/sim.html> – Дата обращения 10.10.2016.

References

1. *Microsoft Robotics Studio – robotehnika dlya vseh* [Microsoft Robotics Studio – robotics for everyone]. Available at: <https://www.bytemag.ru/articles/detail.php?ID=6470>. (accessed 15.11.2016)
2. Gibilisco S. *Concise encyclopedia of robotics*, The McGrawHill Companies, Inc, 2003
3. *Lektsiya 7: Raspredelennoe imitatsionnoe modelirovanie* [Lecture 7: distributed simulation modeling]. Natsionalnyi Otkrytyi Universitet “INTUIT” [National Open University “INTUIT”]. Available at: http://www.intuit.ru/studies/courses/1146/238/lecture/6150?page=1&keyword_content=discrete%20event%20simulation. (accessed 09.2016)

4. Ivanova S.B, Salnicov I. S., Salnicov R. I. Multipurpose robotic computing hardware systems: needs and problems of development. *Problemy iskusstvennogo intellekta* [Problems of Artificial Intelligence], 2015, no. 0 (1), pp. 50-60.
5. Robotics. *3D CAD Design Software | SOLIDWORKS*. Available at: <http://www.solidworks.com/sw/education/robotic-design-resources-students.htm>. (accessed 10.10.2016)
6. *NXT-G: the development environment supplied with Lego Mindstorms, NXT-G*. Génération Robots. Available at: <http://www.generationrobots.com/blog/en/2015/09/nxt-g-the-development-environment-supplied-with-lego-mindstorms-nxt-g>. (accessed 10.10.2016)
7. TRIK Studio, *Всё о ТРИК: TRIK Studio* [Everything about TRIK: TRIK Studio]. Available at: <http://blog.trikset.com/p/trik-studio.html>. (accessed 10.10.2016)
8. Robotics System Toolbox. *MathWorks - Makers of MATLAB and Simulink*. MATLAB & Simulink. Available at: <https://www.mathworks.com/products/robotics>. (accessed 10.10.2016)
9. *anyKode Marilou*. anyKode Marilou - Modeling and simulation environment for Robotics. Available at: <http://doc.anykode.com/frames.html?frmname=topic&frmfile=index.html>. (accessed 21.11.2016)
10. What is Arduino? *Arduino*. Available at: <https://www.arduino.cc/en/Guide/Introduction>. (accessed 10.10.2016)
11. The OROCOS Project. *Departement Werktuigkunde. KU Leuven*. Available at: <https://people.mech.kuleuven.be/~orocos/pub/documentation/rtt/v1.12.x/doc-xml/orocos-overview.html>. (accessed 10.10.2016)
12. Core Components. *ROS.org | Powering the world's robots*. Available at: <http://www.ros.org/core-components>. (accessed 10.10.2016)
13. V-REP User Manual. *Coppelia Robotics V-REP: Create. Compose. Simulate. Any Robot*. Available at: <http://www.coppeliarobotics.com/helpFiles/index.html>. (accessed 01.11.2016)
14. ROBOTC Downloads. *ROBOTC a C Programming Language for Robotics*. Available at: <http://www.robotc.net/download>. (accessed 10.10.2016)
15. Bricx Command Center 3.3. *SourceForge*. Download, Develop and Publish Free Open Source Applications. Available at: <http://bricxcc.sourceforge.net>. (accessed 10.10.2016)
16. Robotino® View. *Festo Didactic*. Available at: <http://www.festo-didactic.com/int-en/services/robotino/programming/robotino-view>. (accessed 10.10.2016)
17. Welcome to Robotics Developer Studio. *MSDN: Learn to Develop with Microsoft Developer Network*. Available at: <https://msdn.microsoft.com/en-us/library/bb648760.aspx>. (accessed 10.10.2016)
18. Beginner: Overview. *Gazebo*. Available at: http://gazebo.org/tutorials?cat=guided_b&tut=guided_b1. (accessed 10.10.2016)
19. What is it? *Algodoo*. Available at: <http://www.algodoo.com/what-is-it>. (accessed 10.10.2016)
20. Urbi Open Source. *Gostai*. Available at: <http://www.gostai.com/products/urbi>. (accessed 10.10.2016)
21. System Overview. *SourceForge*. Download, Develop and Publish Free Open Source Applications. Available at: http://usarsim.sourceforge.net/wiki/index.php/3._System_Overview. (accessed 10.10.2016)
22. Webots. *Cyberbotics*. Available at: <https://www.cyberbotics.com/overview>. (accessed 10.10.2016)
23. robotSim 3D Simulator for NI Robotics Starter Kit - Cogmation Robotics. *National Instruments: Test, Measurement, and Embedded Systems*. Available at: <http://sine.ni.com/nips/cds/view/p/lang/en/nid/209028>. (accessed 10.10.2016)
24. About SARGE. *SourceForge*. Download, Develop and Publish Free Open Source Applications. Available at: <http://sarge.sourceforge.net/page11/page11.html>. (accessed 10.10.2016)
25. Documentation of Iprobots. *Research Network for Self-Organization of Robot Behavior*. Available at: <http://robot.informatik.uni-leipzig.de/software/lprobots-0.4-doc/html/main.html>. (accessed 10.10.2016)
26. SimRobot – A General Physical Robot Simulator and its Application in RoboCup. *Research Network for Self-Organization of Robot Behavior*. Available at: <http://www.informatik.uni-bremen.de/kogrob/papers/rc06-sim.pdf>. (accessed 10.10.2016)
27. SubSim Overview. *Robotics UWA*. Available at: <http://robotics.ee.uwa.edu.au/auv/subsim/doc/About.html>. (accessed 10.10.2016)
28. EyeSim - EyeBot Simulator. *Robotics UWA*. Available at: <http://robotics.ee.uwa.edu.au/eyebot/doc/sim/sim.html>. (accessed 10.10.2016)

RESUME

Y. S. Pikaliyov

Analysis of Existing Simulators for Robotic Systems

Background: Nowadays the automation of various processes is being developed rapidly with robotic tools becoming more and more widely used. The use of simulation in the development of robotic systems is a promising and rapidly developing area in the fields of software support automation and robotics production. This solution is urgent and up-to-date, since it allows reducing costs and speeding up the process of designing robotic systems.

Materials and methods: The review is made on the basis of practical using and developing the robotic systems with the help of simulators.

Results: Advantages and disadvantages of the use of simulators are identified. The classification of actual robotic systems is provided, where advantages and disadvantages of each simulator are described.

Conclusion: The toolkit of a simulator has several advantages: it provides developers with the ability to create or modify robotic systems without discontinuing their production; it allows implementing and optimizing various ideas to carry out tests on a real robotic system with a view to their further realization; it provides tools for analysis of various projects (scenarios) and their sequences; robotic simulator has an intuitive visual environment for product demonstration. The main disadvantage is the problem of robotics modeling with corresponding simulating real robotic systems, environment, and different processes. Also the results of review are displayed in the tables (table 1, table 2, table 3).

Статья поступила в редакцию 10.01.2017.