

Адаптация систем научной визуализации к задачам мониторинга данных от легковесных аппаратных робототехнических устройств

К.В. Рябинин¹, С.И. Чуприна¹

kostya.ryabinin@gmail.com|chuprinas@inbox.ru

¹Пермский государственный национальный исследовательский университет, Пермь, Россия

Данная работа посвящена вопросам адаптации систем научной визуализации к решению задач мониторинга данных, генерируемых в реальном времени легковесными роботизированными устройствами, на примере мультиплатформенной клиент-серверной системы SciVi. Система научной визуализации строится на принципах модельно-ориентированного подхода. Поведение системы управляется онтологической базой знаний, что позволяет без модификации исходного кода ядра этой системы настроить её на решение задач визуализации данных из любой предметной области. За получение данных из стороннего источника и первичную их обработку отвечает серверная часть системы; результат визуализации демонстрируется пользователю на стороне клиента. С целью обеспечения эффективного взаимодействия с роботизированными устройствами в клиентской части SciVi был поддержан режим работы, позволяющий получать данные напрямую из источника без необходимости соединения с сервером. Описываются результаты тестирования системы в этом режиме на примере мониторинга углов курсовых перемещений.

Ключевые слова: научная визуализация, онтологический инжиниринг, системы реального времени, робототехника.

Tackle Lightweight Hardware Robotic Devices Data Monitoring Problems by Means of Scientific Visualization Systems

K. V. Ryabinin¹, S. I. Chuprina¹

kostya.ryabinin@gmail.com|chuprinas@inbox.ru

¹Perm State University, Perm, Russia

This paper is devoted to improving of the multiplatform client-server scientific visualization system SciVi adaptation capabilities to tackle lightweight hardware robotic devices data monitoring problems. SciVi is an ontology based model-driven system, which allows tuning to arbitrary application domain without core source code modification. The SciVi server is responsible for data retrieving and preprocessing, the client demonstrates the final rendering result to the user. The client has been improved to support direct communication with data source without server connection to ensure efficient interaction with external robotics device. This operating mode was tested on the attitude and heading reference system's orientation monitoring.

Keywords: scientific visualization, ontology engineering, real-time systems, robotics.

1. Введение

В настоящее время благодаря удешевлению элементной базы (управляющих и исполнительных элементов, а также источников электропитания), литевых и формовочных материалов (пластиков и силиконов), а также распространению средств быстрого прототипирования (3D-печати, лазерной резки и т. п.) робототехника стала объектом пристального интереса не только крупных научно-исследовательских центров и производств, но и широких масс населения. Например, широкое распространение получили роботы-пылесосы, интерактивные игрушки и т. п. Кроме того, выделилось целое направление хобби-робототехники, предполагающее сборку роботизированных устройств в домашних условиях, а основы робототехники как науки и ремесла были включены в школьную программу в виде факультативных занятий [8].

На базе робототехники существуют различные концепции развития вычислительной техники и инфраструктуры её взаимодействия, например Интернет вещей (англ. *Internet of Things*, *IoT*) [3], умный дом (англ. *Smart Home*) [7], электронный текстиль (англ. *E-Textiles*) [4] и др.

В сфере непромышленной робототехники существует множество доступных по цене и лёгких в освоении аппаратных решений. Наиболее популярные из них стали на сегодняшний день полноценными легковесными платформами для разработки роботизированных устройств. К таким платформам относятся, например, Arduino [6],

предназначенная для решения широкого спектра задач автоматизации, и используемые в основном для решения задач в области Интернета вещей сборки на основе микроконтроллеров линейки ESP [2].

Одновременно с ростом популярности аппаратных средств для построения роботов и автоматизированных систем на их основе, возрос спрос и на программное обеспечение, позволяющее моделировать, программировать, мониторить и контролировать соответствующие устройства. Когда речь идёт о сборе, обработке и интерпретации данных, получаемых от роботизированной системы (например, комплексных параметров состояния жилых и нежилых помещений, снимаемых в реальном времени датчиками умного дома), в роли подобного программного обеспечения могут выступать системы научной визуализации.

Ранее авторами была предложена концепция создания адаптивных мультиплатформенных систем научной визуализации на основе методов и средств онтологического инжиниринга [9]. Эта концепция была воплощена на практике в виде программной системы SciVi.

Целью данной работы является дальнейшее совершенствование адаптационных возможностей систем научной визуализации для решения задач мониторинга легковесных непромышленных роботизированных устройств на примере системы SciVi. Это позволит применять систему SciVi для прототипирования и малой автоматизации целого ряда процессов как в научно-технической, так и в бытовой сфере.

2. Проблема взаимодействия с системами реального времени

Благодаря модельно-ориентированному подходу, лежащему в основе архитектуры системы SciVi, эта система при помощи средств высокоуровневого графического интерфейса может быть адаптирована к стороннему источнику данных без модификации исходного кода её ядра [9]. Взаимодействие с источником данных осуществляется через серверную часть SciVi, тогда как демонстрация результатов визуализации происходит средствами клиентской части. В зависимости от сложности данных и вычислительной мощности клиента, визуализация может происходить как на стороне сервера, так и на стороне клиента, либо быть распределена между этими вычислительными узлами.

Такой подход обеспечивает достаточную универсальность при визуализации данных из произвольных источников. Однако в некоторых частных случаях этот подход может оказаться неоптимальным, например, при взаимодействии с системами реального времени, к которым относится большинство робототехнических устройств.

Согласно стандарту POSIX 1003.1, система реального времени – это система, способная обеспечить требуемый уровень сервиса в определённый промежуток времени, т. е. реагировать в предсказуемое время на непредсказуемое появление внешних событий. При визуализации данных, получаемых от таких систем, время их сбора и рендеринга не должно превышать времени их генерации. Только в этом случае пользователь сможет производить адекватный мониторинг и своевременно реагировать на произошедшие события. Соответственно, наиболее оптимальным оказывается сценарий прямого взаимодействия визуализатора с источником данных, по возможности минуя посредников.

В контексте взаимодействия с легковесными робототехническими платформами речь, как правило, идёт об относительно небольших объёмах данных, подлежащих визуализации. Это обусловлено малой вычислительной мощностью аппаратной части. Так, например, тактовая частота используемых в этих платформах микроконтроллеров обычно не превышает 100 МГц, а объём оперативной памяти чаще всего оказывается менее 1 Мб.

В связи с этим даже в случае, когда в роли клиента системы SciVi выступает маломощное мобильное устройство, система окажется в состоянии обработать и визуализировать весь требуемый объём данных. Следовательно, более оптимальным в этом случае оказывается сбор данных на стороне клиента, минуя сервер.

Для пользователя исключение сервера SciVi из цепочки обработки данных при работе с робототехническими устройствами упрощает работу с системой визуализации, так как делает инфраструктуру вычислительной сети более тривиальной и интуитивно-понятной. Кроме того, этот режим позволяет системе SciVi, развёрнутой, например, на смартфоне, планшетном компьютере или ноутбуке, взаимодействовать с робототехническими устройствами, работающими в режиме точки доступа WiFi.

Для поддержки эффективного взаимодействия системы SciVi с легковесными робототехническими устройствами были выполнены следующие расширения:

1. Настройка на новые виды источников данных.
2. Реализация компонента получения данных напрямую с робототехнического устройства.

3. Настройка системы SciVi на взаимодействие с робототехническим устройством

Настройка системы SciVi на сторонний источник данных осуществляется посредством Web-интерфейса её серверной части. В том случае, если источник данных представлен программно-аппаратным решателем и пользователь предоставляет доступ к исходному коду программной части решателя, процесс настройки оказывается во многом автоматизирован. Система автоматически генерирует синтаксический анализатор для поданного на вход кода решателя на основе онтологии синтаксических конструкций ввода-вывода языков программирования [9]. В результате анализа из этого кода извлекаются описания входных и выходных данных, и пользователю предоставляется высокоуровневый графический интерфейс для выбора интересующих его параметров. В результате такой настройки формируется онтологический профиль источника данных, используемый затем для автоматического построения входной вершины в диаграмме потока данных, используемой в качестве спецификации внутреннего конвейера визуализации [9].

Если исходный код недоступен, пользователь может самостоятельно описать входные и выходные данные посредством высокоуровневого графического Web-интерфейса.

В случае настройки на работу с роботизированным устройством исходным кодом решателя выступает код прошивки соответствующего устройства. Как правило, этот код бывает доступен пользователю, особенно, если речь идёт о робототехнике, используемой для целей образовательного процесса, или о хобби-робототехнике: пользователь либо сам пишет этот код, либо скачивает прошивку для создаваемого им устройства из сети Интернет. Большинство популярных готовых прошивок распространяются в виде исходного кода (как правило, по OpenSource-лицензиям).

Взаимодействие с легковесными встраиваемыми системами осуществляется через проводные и беспроводные интерфейсы. К наиболее популярным проводным интерфейсам относятся I²C, SPI и UART. Беспроводная связь организуется посредством WiFi и Bluetooth. Для каждого способа коммуникации есть типовые операции ввода-вывода, реализованные для наиболее популярных платформ в виде библиотек, ставших стандартом де-факто. К таким библиотекам относятся, например, Serial и WiFi, предназначенные для коммуникации по стандартам RS-232 (UART) и IEEE 802.11 (WiFi) соответственно. Эти библиотеки изначально были реализованы на языке C++ для платформы Arduino, а затем портированы и на другие популярные платформы (например платформы на базе микроконтроллеров линейки ESP).

Основой организации поддержки взаимодействия SciVi с робототехническими платформами стало расширение онтологии синтаксических конструкций ввода-вывода описанием соответствующих вызовов популярных коммуникационных библиотек. Для реализации тестового примера и проверки жизнеспособности подхода были поддержаны программные интерфейсы библиотек Serial и WiFi на языке C++. Для библиотеки WiFi поддержаны функции приёма и передачи данных по протоколам TCP и UDP, при этом используемый протокол сохраняется в онтологическом профиле источника данных в виде метainформации.

Модификации, связанные с обеспечением возможности настройки на соответствующий источник данных (робототехническое устройство), в соответствии с

модельно-ориентированным подходом, свелись к пополнению онтологии синтаксических конструкций ввода-вывода, а также добавлению в набор настроек единственной новой опции – флага использования режима прямого доступа к источнику данных (минуя сервер). Значение этой опции сохраняется в онтологический профиль источника данных в виде метаданных.

Этап настройки по-прежнему осуществляется на стороне сервера SciVi, так как перенос его на сторону клиента для мультиплатформенных систем не оправдан с технической точки зрения. Например, в случае, если клиент работает под управлением ОС iOS, для загрузки в систему файлов исходного кода решателя (с целью осуществления настройки на стороне клиента) потребовалось бы использование сторонних сервисов хранения (таких как Dropbox или Яндекс.Диск).

4. Режим работы системы SciVi без сервера

Для обеспечения поддержки прямого взаимодействия клиента SciVi с роботизированным устройством в состав клиента был добавлен расширяемый модуль связи, названный SciViATRobot, работающий с наиболее популярными средствами коммуникации – UART, WiFi и Bluetooth. Архитектура модуля схематично представлена на рис. 1.

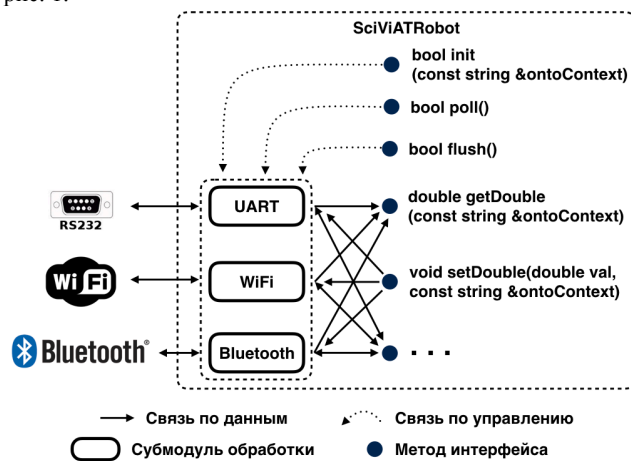


Рис. 1. Схема модуля SciViATRobot.

На программном уровне модуль имеет единообразный интерфейс для всех коммуникационных средств (в связи с чем при необходимости его функциональность может быть расширена). Этот интерфейс включает в себя метод инициализации соединения (init), а также методы запроса и отправки данных всех поддерживаемых базовых типов – целых и вещественных чисел, строк, векторов и т. п. (например, методы getDouble и setDouble для оперирования вещественными числами). Позиции запрашиваемых и отправляемых значений во входном и выходном сообщениях, а также используемый протокол передачи данных (например, для WiFi – TCP или UDP) определяются на основе онтологического профиля источника данных, созданного сервером системы SciVi. Ввод в вывод буферизированы, в связи с чем интерфейсом дополнительно предусмотрены методы опроса внешнего устройства (poll) и отправки сообщений (flush).

В том случае, если при настройке на источник данных пользователем был выбран режим прямого доступа, все входные данные системы визуализации с заданной периодичностью запрашиваются путём обращения к интерфейсу модуля SciViATRobot. Периодичность зависит от скорости передачи данных (декларированной конкретным стандартом связи) и скорости перерисовки графической сцены.

Следует отметить, что реализация и возможности этого модуля зависят от программно-аппаратной платформы клиента в силу различий в наборах коммуникационных средств (например, на современных мобильных устройствах отсутствует аппаратный UART).

5. Пример мониторинга системы реального времени

В качестве тестового примера системы реального времени на основе робототехнической платформы, авторами была собрана курсоверткаль (англ. *Attitude and Heading Reference System, AHRS*) [10] – прибор для определения углов Тейта-Брайана: крена (англ. *Roll*), тангажа (англ. *Pitch*) и рыскания (англ. *Yaw*). Основной элементной базой этого устройства являются следующие компоненты:

1. Трёхосевой гироскоп марки MPU3200, служащий для определения углов крена и тангажа.
2. Трёхосевой акселерометр марки ADXL345, служащий для компенсации погрешности измерений гироскопа.
3. Трёхосевой магнитометр марки HMC5883L, служащий для определения угла рыскания (на основе известных углов крена и тангажа).
4. WiFi-модуль ESP-07 на базе микроконтроллера ESP8266, служащий для управления устройством и передачи данных в эфир.

Устройство является полностью автономным (источником питания служат элементы типа AAA) и настроено на коммуникацию с внешними системами посредством сети WiFi, работая в режиме точки доступа.

Прошивка устройства была написана авторами с использованием библиотек [1], реализованных сообществом ESP8266-разработчиков. Микроконтроллер ESP8266 – самый популярный представитель линейки ESP, имеющий несколько вариантов компоновки, в частности, модель ESP-07 (производимая компанией AI-Thinker), отличающаяся очень малым размером, но, при этом, наличием встроенной керамической WiFi-антенны, обеспечивающей стабильную связь на расстоянии до 300 метров. Для него силами международного сообщества разработчиков сформирована обширная программная инфраструктура и удобный инструментальный для написания прошивок.

Технические характеристики данного микроконтроллера позволяют решать с его помощью достаточно широкий спектр задач, включая достаточно сложную обработку данных, получаемых с периферических сенсоров. Его тактовая частота составляет 80 МГц, объём оперативной памяти – 64 Кб для кода и 96 Кб для данных (в соответствии с Гарвардской архитектурой, на принципах которой построен этот микроконтроллер, исполняемый код и данные находятся на различных запоминающих устройствах). На аппаратном уровне поддерживаются интерфейсы SPI, I²C, I²S и UART, а также аналого-цифровой преобразователь разрядностью 10 бит. Для подключения периферических сенсоров и других устройств имеется 10 свободных пинов ввода-вывода.

Созданная для ESP-07 прошивка включает в себя функции опроса гироскопа, акселерометра и магнетометра (связанных с микроконтроллером по интерфейсу I²C), определение по полученным данным углов крена, тангажа и рыскания, и сглаживание осцилляций этих углов (неизбежно возникающих в силу случайных шумов на сенсорах) при помощи фильтра Калмана [5]. Кроме того, предусмотрена функция выставки курсоверткаль – задания текущей ориентации устройства в качестве отсчётной (ориентации, при которой выдаваемые устройством углы одновременно равны нулю). Выставка

осуществляется нажатием тактовой кнопки. Радианные меры углов крена, тангажа и рыскания (с учётом выставки) с частотой 50 Гц передаются по протоколу UDP в сеть (в виде троек чисел с плавающей точкой двойной точности), для чего используется библиотека WiFi, портированная сообществом на ESP8266.

На внутреннем уровне текущая ориентация представляется в виде кватерниона q_c , который строится по обработанным фильтром Калмана значениям крена, тангажа и рыскания, получаемым с соответствующих сенсоров. Такой подход упрощает выставку: в момент нажатия соответствующей тактовой кнопки текущий кватернион q_c присваивается кватерниону q_r , задающему отсчётную ориентацию: $q_r = q_c$. Ориентация с учётом выставки (ориентация, углы которой отправляются в сеть) задаётся кватернионом $q_s = q_c \cdot q_r^{-1}$.

Настройка системы SciVi на мониторинг данных, генерируемых собранным устройством, была осуществлена путём подачи на вход системы исходного кода прошивки этого устройства (а также указания режима прямого доступа к источнику данных). В результате автоматически сформировалось описание источника данных как триплета вещественных чисел, задающих крен, тангаж и рыскание. На этапе настройки графической сцены углы ориентации были связаны с 3D-моделью, которая в процессе визуализации принимает получаемую от курсоверткали ориентацию. Диаграмма потока данных, используемая на клиенте SciVi, а также соответствующий результат визуализации представлены на рис. 2.

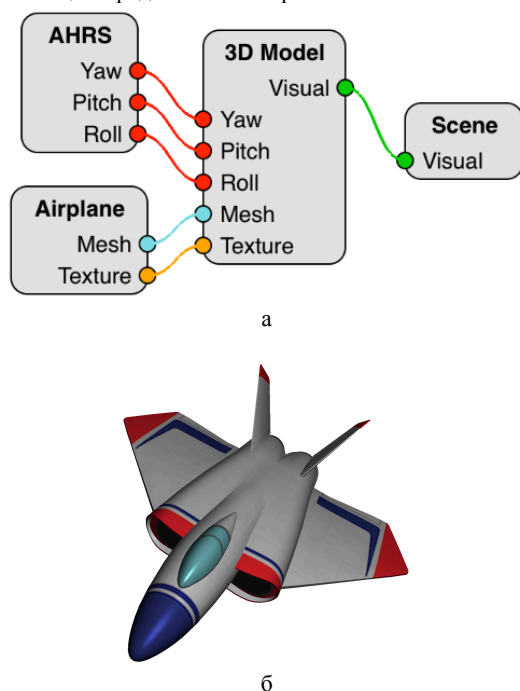


Рис. 2. Диаграмма потока данных (а) и результат визуализации ориентированной 3D-модели по данным с курсоверткали (б) на клиенте SciVi.

Вершина «AHRS» на диаграмме потока данных обозначает точку связи с курсоверткалью. Вершина «3D Model» отвечает за формирование визуального образа трёхмерной модели, имеющей ориентацию в пространстве (выраженную углами Тейта-Брайана), каркас и текстуру (для упрощения диаграммы другие параметры модели, такие, например, как положение в пространстве или свойства материала, были выставлены в значения по умолчанию и удалены из множества входных параметров на уровне онтологии визуальных объектов). Вершина «Airplane» определяет каркас и текстуру модели самолёта.

Вершина «Scene» служит для формирования графической сцены из поданных ей на вход визуальных объектов.

6. Заключение

В данной работе описаны результаты успешной адаптации системы SciVi к решению задач мониторинга данных, получаемых от легковесных робототехнических устройств, эффективного с точки зрения скорости и удобства настройки. Для этого в состав клиентской части системы был включён программный модуль прямой связи с роботизированным устройством, не нуждающийся во время своей работы в подключении к серверу SciVi.

Первичная адаптация к роботизированному устройству по-прежнему производится на стороне сервера, так как включает анализ исходного кода прошивки устройства, однако после этого этапа подключение к серверу более не требуется. Так, например, если сервер доступен в виде сервиса (англ. *Software as a Service, SaaS*) в сети Интернет, пользователь не должен будет заботиться о его развёртывании и обслуживании. Для формирования онтологического профиля соответствующего роботизированного устройства пользователю достаточно лишь кратковременного подключения к серверу для выполнения настройки. Затем можно отсоединиться от сети Интернет и, подключившись к локальной сети (возможно, созданной самим роботизированным устройством), осуществлять мониторинг аппаратной системы в реальном времени.

Взаимодействие системы SciVi с роботизированными устройствами было протестировано на примере мониторинга углов курсоверткали. В будущем планируется также отладить обратную связь SciVi с внешними устройствами, что позволит использовать её в режиме монитора, совмещённого с пультом управления робототехнической системой.

Следует также отметить, что в режиме работы без сервера система SciVi может эффективно взаимодействовать и с различными сторонними Web-сервисами, развёрнутыми в сети Интернет, например, сервисами погоды, курсов валют и т. п. Таким образом, разработанный режим в значительной степени увеличивает сферы её применимости.

7. Литература

- [1] Arduino IDE для ESP8266 // ESP8266 - Сообщество разработчиков. – 2014 [Электронный ресурс]. URL: <https://esp8266.ru/arduino-ide-esp8266/> (дата обращения 19.05.2017).
- [2] Brown E. Linux and Open Source Hardware for IoT // Linux.com. – 2016 [Электронный ресурс]. URL: <https://www.linux.com/news/linux-and-open-source-hardware-iot> (дата обращения 19.05.2017).
- [3] Brown E. Who Needs the Internet of Things? // Linux.com. – 2016 [Электронный ресурс]. URL: <https://www.linux.com/news/who-needs-internet-things> (дата обращения 19.05.2017).
- [4] Jamadar S. Applications of Smart and Interactive Textiles // Textile Learner. – 2013 [Электронный ресурс]. URL: <http://textilelearner.blogspot.ru/2013/04/applications-of-smart-and-interactive.html> (дата обращения 19.05.2017).
- [5] Kalman R.E. A New Approach to Linear Filtering and Prediction Problems // Journal of Basic Engineering. – 1960. – No. 82. – 11 p.
- [6] Kushner D. The Making of Arduino // IEEE Spectrum. – 2011 [Электронный ресурс]. URL: <https://www.ieee.org/spectrum/2011/05/arduino>

<http://spectrum.ieee.org/geek-life/hands-on/the-making-of-arduino> (дата обращения 19.05.2017).

- [7] Yi Man Li R., Ching Yu Li H., Kei Mak C., Beiqi Tang T. Sustainable Smart Home and Home Automation: Big Data Analytics Approach // International Journal of Smart Home. – 2016. – Vol. 10, No. 8. – PP. 177–198.
- [8] Жупиков Ю.Ю. Робототехника как фактор формирования исследовательской компетентности учащихся в рамках внеурочной деятельности // Сборник тезисов VI Всероссийской конференции «Современное технологическое обучение: от компьютера к роботу». – СПб., 2016. – С. 18–19.
- [9] Рябинин К.В., Чуприна С.И., Бортников А.Ю. Автоматизация настройки систем научной визуализации на специфику разнообразных источников данных // Научная визуализация. – М.: Национальный исследовательский ядерный университет МИФИ, 2016. – К. 4, Т. 8, №4. – С. 1–14.
- [10] Соловьёв В.И., Шабалов П.Г. Инерциальные навигационные системы // Учеб. пособие; Самар. . гос. нац. исслед. аэрокосм. ун-т. им. С.П. Королёва. – Самара: СГАУ, 2011. – 74 с.

Об авторах

Константин Валентинович Рябинин, к.ф.-м.н., доцент кафедры математического обеспечения вычислительных систем Пермского государственного национального исследовательского университета. Его электронный адрес kostya.ryabinin@gmail.com.

Светлана Игоревна Чуприна, к.ф.-м.н., заведующая кафедрой математического обеспечения вычислительных систем Пермского государственного национального исследовательского университета. Её электронный адрес chuprinas@inbox.ru.