

DOI: 10.15593/2499-9873/2020.4.06

УДК 004.89

А.И. Мейер

Национальный исследовательский университет
«Высшая школа экономики», Пермский филиал, Пермь, Россия

РАЗРАБОТКА КОНЦЕПТУАЛЬНОЙ АРХИТЕКТУРЫ УНИВЕРСАЛЬНОЙ МНОГОМОДУЛЬНОЙ ПЛАТФОРМЫ АВТОНОМНОГО ПОРТАТИВНОГО ТРАНСПОРТНОГО СРЕДСТВА

Данная статья является логическим продолжением прошлой работы, посвященной литературному обзору в сфере портативных автономных транспортных средств с функцией передвижения в неопределенном динамическом ландшафте. В рамках нее был выбран наилучший алгоритм для реализации функции движения, а также рассмотрены примеры виртуальных платформ (сред), программных пакетов и существующих некоммерческих проектов. В данной работе исследование было продолжено и конкретизировано. Решено сконцентрироваться на универсальных многомодульных платформах, с помощью которых можно унифицировать производство автономных портативных транспортных средств. Представлена разработанная концептуальная архитектура такой платформы. Нотацией (языком) моделирования архитектуры был выбран Unified Modeling Language (UML), а основными диаграммами, которые необходимо описать для полноценного концептуального описания, – диаграмма Use Case (прецедентов), диаграмма развертывания и диаграмма последовательностей.

Ключевые слова: нейронная сеть, автопилот, автономное управление, киберфизические системы, портативные транспортные средства, глубокое обучение, виртуальные платформы, архитектура.

A.I. Meyer

National Research University Higher School of Economics,
Perm Branch, Perm, Russian Federation

THE DEVELOPMENT OF CONCEPTUAL ARCHITECTURE OF SELF-DRIVING PORTABLE VEHICLE'S UNIVERSAL MULTI-MODULAR PLATFORM

This article is the logical sequel of my previous one, which deals with literature review in terms of self-driving portable vehicles with functionality of movement inside undetermined dynamic environment. That research paper describes the process and results of the best movement algorithm selection. Also it contains review of available virtual platforms, frameworks and existing non-commercial projects. In this article the research was continued and made more specific. I decided to concentrate on universal multi-module platforms, which can be used to standardize the production of self-driving portable vehicles. This study covers the results of conceptual architecture development. Unified Modeling Language

(UML) was selected as the notation for it. Also there are three diagrams, that were defined to illustrate this architecture: Use Case diagram, sequence diagram and deployment diagram.

Keywords: neural network, autopilot, autonomous control, cyber-physical systems, portable vehicles, deep learning, virtual platforms, architecture.

Введение

С недавнего времени общество заинтересовалось в разработке автономных систем с искусственным интеллектом. В частности, к ним можно отнести роботов, умные вещи, умные дома и т.д. Особое внимание занимают транспортные средства с автопилотом (подсистемой, самостоятельно выполняющей задачи управления). В работе [1] было упомянуто мнение, согласно которому обоснованием высокого интереса к этому виду киберфизических систем является то, что внедрение автономного транспорта является одной из первых и в то же время глубочайшей интеграцией роботов для личного использования, в результате чего их развитие может существенно повысить доверие человека к машине. Это подкрепляется тем фактом, что когда человек, например, передвигается в автомобиле с системой автопилота, то он доверяет искусственному интеллекту свою жизнь и здоровье.

Данная статья является логическим продолжением моей прошлой работы, указанной в источнике [2]. В ней был проведен литературный обзор алгоритмов, программных пакетов и виртуальных платформ, которые пригодны для разработки автономного портативного транспортного средства с функцией передвижения в неопределенном динамическом ландшафте. Также был осуществлен сравнительный анализ алгоритмов, пригодных для реализации функции передвижения, в рамках которого DQN (глубокое Q-обучение), впервые представленный в работе [3], был выбран наилучшим.

В качестве предложения было решено обратить внимание на разработку концептуальной архитектуры универсальной многомодульной платформы, на базе которой можно будет собрать и использовать подобный транспорт, так как именно она даст основное понимание полезности и базовых особенностей реализации задуманной системы. Архитектура должна соответствовать выдвинутым в статье [2] выводам. Также для проверки выбранного алгоритма необходимо провести эксперимент с его использованием. Для этого может подойти одна из виртуальных платформ, упомянутых в предыдущей работе:

- CARLA [4];
- Autoware [5];
- AirSim [6];
- Gazebo [7];
- Euro Truck Simulator 2 [8];
- Multi Theft Auto [9].

При проведении эксперимента возможно использование программных пакетов, также описанных в источнике [2], параллельно с виртуальными платформами:

- Torch [10];
- OpenCV [11].

В данной статье внимание будет сконцентрировано именно на архитектуре. Выбор виртуальной платформы для проведения эксперимента будет осуществлен в дальнейшем. Исходя из этого, *целью* данной работы является разработка концептуальной архитектуры универсальной многомодульной платформы автономного портативного транспортного средства.

Для достижения поставленной цели требуется выполнить ряд следующих задач:

- выбрать нотацию и составляющие части концептуальной архитектуры;
- разработать составные части архитектуры в соответствии с выбранным языком.

1. Язык и составляющие части архитектуры

Для описания концептуальной архитектуры необходимо определить следующие элементы рассматриваемой системы:

- набор выполняемых рассматриваемой системой задач;
- программно-аппаратные части системы;
- работа и взаимодействие элементов системы.

В статье [12] были охарактеризованы и проанализированы следующие графические нотации, которые можно использовать при выполнении текущей задачи:

- IDEF (IDEF0, IDEF3);
- DFD;
- EPC;

- ГАС;
- BPMN;
- UML.

В качестве языка моделирования, подходящего под данную задачу, необходимо использовать Unified Modeling Language (UML), так как в отличие от остальных перечисленных нотаций он поддерживает объектно-ориентированное описание архитектуры, что, на мой взгляд, делает диаграммы мнемоническими и доступными для широкого круга людей.

Для общего описания платформы с использованием нотации UML необходимо составить представление об ее структуре, поведении и взаимодействиях составляющих компонентов [13]. Для этого подходят следующие виды диаграмм:

- диаграмма прецедентов (поведение);
- диаграмма развертывания (структура);
- диаграмма последовательностей (взаимодействие).

Диаграмма прецедентов позволяет выделить основные сущности системы, а также определить круг задач, которые выполняет каждая из них [14]. Благодаря данной диаграмме можно определить границы основной функциональности рассматриваемой системы.

Диаграмма развертывания позволяет создать представление о системе на физическом уровне [15]. В схеме приводится информация о платформе, вычислительных мощностях и иных компонентах системы. Конечно, в силу концептуальности текущей архитектуры данная диаграмма имеет абстрактный характер.

Диаграмма последовательностей строится на основе выбранного алгоритма передвижения в системе и отражает порядок и содержание процесса выбора агентом действия в той или иной ситуации. Данный вид диаграмм позволяет описать алгоритм как передачу сообщений между логическими элементами системы [16], что положительно влияет на воспринимаемость схемы. Следовательно, такая диаграмма хорошо подходит при использовании системного подхода.

Описанных выше UML-диаграмм, на мой взгляд, достаточно, чтобы дать концептуальное описание архитектуре системы автономного портативного транспортного средства. С помощью них будут рассмотрены как состав системы, так и ее работа и взаимодействие между составными частями в ней.

2. Поведение

Диаграмма прецедентов (рис. 1) содержит пять актеров и семь прецедентов. Стоит начать рассмотрение с актеров, основными из которых являются:

- модуль;
- встроенный компьютер;
- датчик.

Базовыми принципами задуманной платформы, исходя из названия, является ее многомодульность и универсальность, что означает, что она должна позволять создавать портативные транспортные средства с помощью сборки унифицированных составляющих деталей в одну систему. Данными деталями и являются модули. На уровне концепции предполагается, что они должны делиться на три вида:

- исполнительные звенья;
- части несущей конструкции;
- модули общего назначения.

Исполнительные звенья имеют непосредственную связь с встроенным компьютером, о котором речь пойдет чуть позже. Связь осуществляется с помощью цифрового интерфейса, с помощью которого идет обмен данными между ЭВМ и модулем. Данные, полученные исполнительным звеном, используются им для смены собственного состояния в соответствии с полученной задачей.

В случае с **частями несущей конструкции** можно провести аналогию с рамой полноразмерного автомобиля, на которой прямо или косвенно держатся все его составляющие элементы. В случае с задуманной платформой предполагается, что с использования данных частей будет начинаться сборка конкретного экземпляра портативного транспорта. Данные детали должны иметь универсальные крепления, которые можно будет использовать для присоединения прочих модулей.

Все остальные детали, которые не соединены напрямую с встроенным компьютером, не являются частью несущей конструкции, относятся к **модулям общего назначения** (на диаграмме прецедентов они обозначены как «модуль»). В качестве примера можно снова привести полноразмерные автомобили. Предположим, что руль является интерфейсом между встроенным компьютером и рулевым управлением. Сам

руль условно соединен напрямую с рулевой рейкой. В данном случае рейка будет исполнительным звеном, а последующие элементы рулевого управления, включая колеса, будут модулями общего назначения.

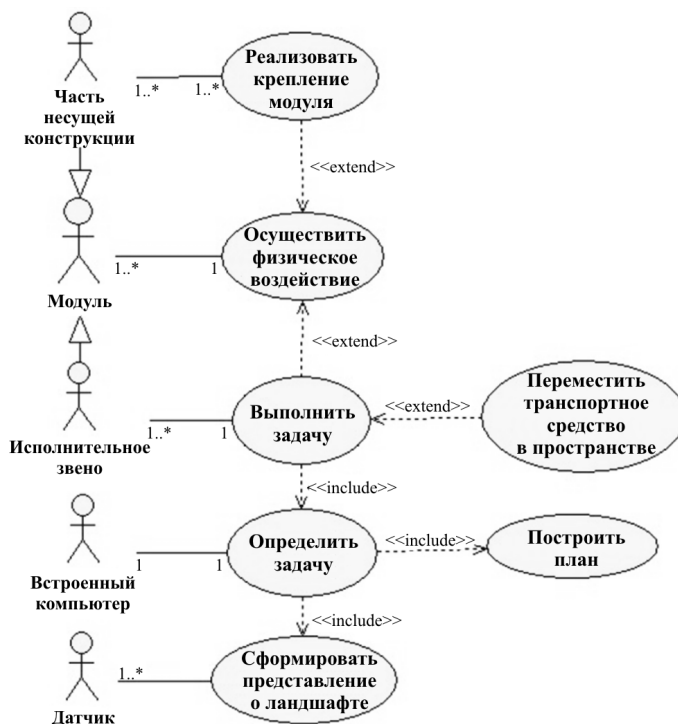


Рис. 1. Диаграмма прецедентов

Встроенный компьютер является источником вычислительных мощностей системы и имеет минимальные габариты. Данный элемент платформы необходим для реализации всех функций портативного транспортного средства, в том числе передвижения в неопределенном динамическом ландшафте. Он является главной единицей, делающей транспортное средство автономным, так как этот узел осуществляет планирование и подачу команд исполнительным звеньям. Иными словами, можно сказать, что компьютер является «шиной» с точки зрения использованных в конструкции транспортного средства модулей. Предполагается, что как минимум алгоритм передвижения реализуется через нейросеть, которая хранится в физической памяти компьютера. Алгоритмы вторичных функций остаются неопределенными. Но существует вероятность, что в нейросеть можно заложить весь функционал

системы и ее модулей, так как сам метод глубокого Q-обучения, выбранный ранее, может работать в общей парадигме. Также встроенный компьютер имеет интерфейсы, пригодные для обмена данными с модулями, а также модуль беспроводной связи (например, для удаленного управления через какое-либо устройство).

Поскольку платформа является замкнутой системой, необходимы дополнительные элементы, формирующие ее внутреннее состояние. Ими являются *датчики*, которые непрерывно считывают информацию об окружении, тем самым формируя сенсорные данные, которые поступают на вход компьютеру. Он же, в свою очередь, уже использует их для планирования и постановки задач.

3. Структура

Диаграмма развертывания, приведенная на рис. 2, содержит все элементы, продемонстрированные на диаграмме вариантов использования. Согласно нотации диаграммы, особое внимание уделено содержанию и связям составляющих единиц платформы. Стоит отметить упомянутое ранее условие концептуальности и основательности данной диаграммы: она лишь демонстрирует самые необходимые связующие части системы.



Рис. 2. Диаграмма развертывания

Встроенный компьютер имеет две составляющие:

- физическое хранилище;
- модуль беспроводной связи.

Примером **физического хранилища** можно назвать USB-, SSD- или HDD-накопитель. Оно обеспечивает работоспособность компьютера, выполняя задачу хранения данных, необходимых для запуска и функционирования портативного транспортного средства. Прежде всего, это нейронная сеть. К сожалению, на данном этапе разработки нельзя точно сказать, будет ли это уже обученная на виртуальной платформе нейросеть или же изначально она будет представлять собой базовый вариант, которому нужно будет пройти обучение с подкреплением согласно выбранному алгоритму.

Модуль беспроводной связи необходим для того, чтобы у экземпляра автономного транспорта, собранного на базе рассматриваемой платформы, был канал приема/передачи данных между устройством управления (например, специальным пультом управления) и самим автономным транспортом. Кроме того, данный элемент может быть полезен для отслеживания рабочих параметров портативного транспортного средства или синхронизации с веб-службами (например, отслеживание мирового времени).

Часть несущей конструкции является элементом корневой составляющей физической оболочки автономного портативного транспортного средства. Это отчетливо видно на диаграмме развертывания, где все вращается вокруг данной единицы платформы. Именно данная секция путем механических соединений связывает все элементы системы воедино.

Исполнительное звено связано с встроенным компьютером через проводную связь (к примеру, это может быть USB-канал) и, как уже было упомянуто при описании поведенческой составляющей системы, совершает через нее обмен информацией, заключающийся в приеме задач от компьютера и возврате текущего состояния и результатов обработки полученных команд обратно. Также оно имеет физическое крепление с основой портативного транспортного средства через связь с частью несущей конструкции.

Модуль, в отличие от исполнительного звена, не связан напрямую с встроенным компьютером, так как он не имеет какого-либо интерфейса для приема команд. Вместо этого данная единица имеет связь

с исполнительным звеном, которое может передавать физическое воздействие на модуль (иногда – косвенно через другие модули) благодаря подвижному соединению с ним.

Все **датчики** каждого экземпляра портативного транспорта крепятся к несущей конструкции. Также, по аналогии с исполнительным звеном, они имеют проводную связь с встроенным компьютером. Отличие в том, что датчики отправляют сенсорные и рабочие данные, необходимые для дальнейшей постановки задач, встроенному компьютеру, но при этом не являются обработчиками данных и, следовательно, не требуют обратной связи.

4. Взаимодействие

Диаграмма последовательностей, приведенная на рис. 3, содержит основной, пользовательский алгоритм работы платформы. На диаграмме к уже описанным элементам системы добавился *пользователь*, так как именно он задает цель, которую должно выполнять автономное портативное транспортное средство.



Рис. 3. Диаграмма последовательностей

Как уже было продемонстрировано в структурной диаграмме, **нейросеть** является частью встроенного компьютера, так как она хранится в его физическом хранилище, поэтому их взаимодействие является условным и отражает лишь необходимые концептуальные детали алгоритма. Поскольку алгоритм глубокого Q-обучения, выбранный в данной работе, поддерживает широкий охват стека операций в алгоритме принятия решений автопилота, искусственная нейронная сеть, используемая в рассматриваемой платформе, способна взять на себя

большую часть операций (выделение характеристик, понимание, планирование и т.д.). Исходя из этого, нейросеть принимает на вход сырые сенсорные данные и возвращает в качестве результата набор задач для исполнительных звеньев, который является наилучшей комбинацией для достижения поставленной пользователем цели в момент использования транспортного средства с учетом настоящего внутреннего состояния системы.

После получения набора задач встроенный компьютер, используя проводную связь, отправляет команды *исполнительным звеньям*, которые, в свою очередь, благодаря наличию подвижных соединений с *модулями* управляют ими. Если модули в связке отсутствуют, то компьютер управляет лишь непосредственно исполнительным звеном. После выполнения задачи исполнительное звено возвращает результат и рабочие значения, полученные с его внутренних датчиков.

После завершения вышеописанных операций алгоритм возвращается к началу цикла и снова отправляет сенсорные данные нейронной сети. Цикл продолжается до тех пор, пока не будет достигнута заданная для автономного транспортного средства цель.

Стоит отметить, что на диаграмме опущены некоторые условности. Например, возможно предположить существование не одной цели, а упорядоченного набора. Также эти цели могут ставиться автоматически без пользователя (например, в цикле или исходя из детерминированной логики, заложенной разработчиком единицы транспортного средства). Продемонстрированная диаграмма последовательностей не демонстрирует процесс обучения системы автопилота, так как на данном этапе непонятно, в какой момент оно будет проходить и в каких условиях. С другой стороны, данный процесс будет иметь немало общего с тем алгоритмом, который представлен на рассматриваемой схеме взаимодействия.

Заключение

В рамках данной статьи была разработана концептуальная архитектура рассматриваемой системы на основе языка UML. Она состоит из трех диаграмм: диаграммы прецедентов, диаграммы последовательностей и диаграммы развертывания.

Факт достижения поставленной в рамках данной статьи цели может быть значим в дальнейшей постановке эксперимента на базе вир-

туальной платформы с целью проверки всех полученных на момент написания статьи результатов. Также они могут быть полезны другим людям, которые ведут исследования в данной сфере.

Список литературы

1. Bhutani S. MIT 6.S094: Deep Learning for Self-Driving Cars 2018 Lecture 1 Notes. – URL: <https://hackernoon.com/mit-6-s094-deep-learning-for-self-driving-cars-2018-lecture-1-notes-807be1a50893> (accessed 12 June 2020).

2. Мейер А.И. Алгоритмы, программные пакеты и платформы автономного управления портативным транспортным средством в условиях неопределенного динамического ландшафта: литературный обзор // Прикладная математика и вопросы управления. – 2019. – № 2. – С. 62–83.

3. Playing Atari with Deep Reinforcement Learning / V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, M. Riedmiller // NIPS Deep Learning Workshop, 9th December 2013, Lake Tahoe, USA. – USA, 2013. – 9 p. – URL: <https://arxiv.org/pdf/1312.5602v1.pdf> (accessed 12 June 2020).

4. CARLA: An Open Urban Driving Simulator / A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, V. Koltun // 1st Conference on Robot Learning (CoRL 2017), 3rd October 2017, Mountain View, USA. – USA, 2017. – 16 p. – URL: <https://arxiv.org/pdf/1711.03938.pdf> (accessed 12 June 2020).

5. Autoware on Board: Enabling Autonomous Vehicles with Embedded Systems / S. Kato, S. Tokunaga, Yu. Maruyama, S. Maeda, M. Hirabayashi, Y. Kitsukawa, A. Monroy, T. Ando, Yu. Fujii, T. Azumi // 2018 9th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS), 11–13 April, 2018, Porto, Portugal. – Los Alamitos, California; Washington; Tokyo: IEEE, 2018. – P. 287–296. DOI 10.1109/ICCPS.2018.00035

6. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles / S. Shah, D. Dey, C. Lovett, A. Kapoor // 11th conference on Field and Service Robotics Conference 2017 (FSR 2017), 12–15 September 2017, Zürich, Switzerland. – Zürich, Switzerland, 2017. – 14 p. – URL: <https://arxiv.org/pdf/1705.05065.pdf> (accessed 12 June 2020).

7. Koenig N., Howard A. Design and Use Paradigms for Gazebo, An Open-Source Multi-Robot Simulator // 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566). 28 September – 2 October 2004, Sendai, Japan. – Los Alamitos, California; Washington; Tokyo: IEEE, 2004. – P. 2149–2154. DOI 10.1109/IROS.2004.1389727

8. Comiskey C. The Very Best Euro Truck Simulator 2 Mods. – URL: <https://www.nvidia.com/en-us/geforce/news/the-very-best-euro-truck-simulator-2-mods/> (accessed 12 June 2020).

9. Moss R. Why A Million People Still Play Multiplayer Grand Theft Auto: San Andreas Every Month. – URL: <https://www.rockpapershotgun.com/2016/09/15/why-a-million-people-still-play-multiplayer-grand-theft-auto-san-andreas-every-month/> (accessed 12 June 2020).

10. Collobert R., Kavukcuoglu K., Farabet C. Torch7: A Matlab-like Environment for Machine Learning // BigLearn, NIPS Workshop, 16–17 December 2011, Sierra Nevada, Spain. – Sierra Nevada, Spain, 2011. – 6 p. – URL: https://infoscience.epfl.ch/record/192376/files/Collobert_NIPSWORKSHOP_2011.pdf (accessed 12 June 2020).

11. Обработка изображений с помощью OpenCV / Г.Б. Гарсия, О.Д. Суарес, Х.Л. Эспиноса Аранда [и др.] – М.: Изд-во ДМК Пресс, 2016. – 210 с.

12. Анализ графических нотаций для имитационного моделирования бизнес-процессов предприятия [Электронный ресурс] / О.П. Аксенова, К.А. Аксенов, А.С. Антонова, Е.Ф. Смолий // Современные проблемы науки и образования. – 2013. – № 4. – URL: <http://www.science-education.ru/ru/article/view?id=9745> (дата обращения: 12.06.2020).

13. Бабич А.В. Виды диаграмм UML [Электронный ресурс]. – URL: <https://www.intuit.ru/studies/courses/1007/229/lecture/5954> (дата обращения: 09.06.2020).

14. Бабич А.В. Диаграммы прецедентов: крупным планом [Электронный ресурс]. – URL: <https://www.intuit.ru/studies/courses/1007/229/lecture/5962> (дата обращения: 16.03.2020).

15. Леоненков А.В. Элементы графической нотации диаграммы развертывания [Электронный ресурс]. – URL: <https://www.intuit.ru/studies/courses/32/32/lecture/1024> (дата обращения: 16.03.2020).

16. Бабич А.В. Диаграммы взаимодействия: крупным планом [Электронный ресурс]. – URL: <https://www.intuit.ru/studies/courses/1007/229/lecture/5960> (дата обращения: 16.03.2020).

References

1. Bhutani S. MIT 6.S094: Deep Learning for Self-Driving Cars 2018 Lecture 1 Notes, available at: <https://hackernoon.com/mit-6-s094-deep-learning-for-self-driving-cars-2018-lecture-1-notes-807be1a50893> (accessed: 01 June 2019).

2. Meyer A.I. Methods, Frameworks and Virtual Platforms in Domain of Portable Self-Driving Vehicles Control Adjusted for Undetermined Dynamic Environment Conditions: Literature Review. *Applied Mathematics and Control Sciences*, 2019, no. 2, pp. 62-83.

3. Mnih V., Kavukcuoglu K., Silver D., Graves A., Antonoglou I., Wierstra D., Riedmiller M. Playing Atari with Deep Reinforcement Learning. NIPS Deep Learning Workshop. 9th December 2013, Lake Tahoe, USA, 2013, 9 p, available at: <https://arxiv.org/pdf/1312.5602v1.pdf> (Accessed 12 June 2020).

4. Dosovitskiy A., Ros G., Codevilla F., Lopez A., Koltun V. CARLA: An Open Urban Driving Simulator. 1st Conference on Robot Learning (CoRL 2017), 3rd October 2017, Mountain View, USA, 2017, 16 p, available at: <https://arxiv.org/pdf/1711.03938.pdf> (Accessed 12 June 2020).

5. Kato S., Tokunaga S., Maruyama Yu., Maeda S., Hirabayashi M., Kitsukawa Y., Monroy A., Ando T., Fujii Yu., Azumi T. Autoware on Board: Enabling Autonomous Vehicles with Embedded Systems. 2018 9th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS), 11–13 April, 2018, Porto, Portugal. Los Alamitos, California ; Washington ; Tokyo: IEEE, 2018, pp. 287-296. DOI 10.1109/ICCPS.2018.00035.

6. Shah S., Dey D., Lovett C., Kapoor A. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. 11th conference on Field and Service Robotics conference 2017 (FSR 2017). 12-15 September 2017, Zürich, Switzerland, 2017, 14 p, available at: <https://arxiv.org/pdf/1705.05065.pdf> (Accessed 12 June 2020)

7. Koenig N., Howard A. Design and Use Paradigms for Gazebo, An Open-Source Multi-Robot Simulator. 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566). 28 September - 2 October 2004, Sendai, Japan. Los Alamitos, California; Washington; Tokyo, IEEE, 2004, pp. 2149-2154. DOI 10.1109/IROS.2004.1389727.

8. Comiskey C. The Very Best Euro Truck Simulator 2 Mods, available at: <https://www.nvidia.com/en-us/geforce/news/the-very-best-euro-truck-simulator-2-mods/> (Accessed 12 June 2020).

9. Moss R. Why A Million People Still Play Multiplayer Grand Theft Auto: San Andreas Every Month, available at: <https://www.rockpapershotgun.com/2016/09/15/why-a-million-people-still-play-multiplayer-grand-theft-auto-san-andreas-every-month/> (Accessed 12 June 2020).

10. Collobert R., Kavukcuoglu K., Farabet C. Torch7: A Matlab-like Environment for Machine Learning // BigLearn, NIPS Workshop, 16–17 December 2011, Sierra Nevada, Spain. – 2011. – 6 p. https://infoscience.epfl.ch/record/192376/files/Collobert_NIPSWORKSHOP_2011.pdf (Accessed 12 June 2020).

11. García G.B., Suarez O.D., Aranda J.L.E., Tercero J.S., Gracia I. S. Learning Image Processing with OpenCV. Birmingham, Packt Publishing, 2015, 232 p.

12. Aksenova O.P., Aksenov K.A., Antonova A.S., Smolij E.F. The Analysis of Graphical Notations for Simulation of Enterprise Business Processes. *Modern problems of science and education*, 2013, no. 4, available at: <http://www.science-education.ru/ru/article/view?id=9745> (Accessed 12 June 2020)

13. Babych A.V. UML Diagram Types, available at: <https://www.intuit.ru/studies/courses/1007/229/lecture/5954> (Accessed 09 June 2020).

14. Babych A.V. Use Case Diagrams: Close-up, available at: <https://www.intuit.ru/studies/courses/1007/229/lecture/5962> (Accessed 16 March 2020).

15. Leonenkov A.V. Elements of Deployment Diagram Graphical Notation, available at: <https://www.intuit.ru/studies/courses/32/32/lecture/1024> (Accessed 16 March 2020).

16. Babych A.V. Interaction Diagrams: Close-up, available at: <https://www.intuit.ru/studies/courses/1007/229/lecture/5960> (Accessed 16 March 2020).

Статья получена: 15.06.2020

Статья принята: 16.11.2020

Сведения об авторе

Мейер Артем Игоревич (Пермь, Россия) – магистрант, Национальный исследовательский университет «Высшая школа экономики», Пермский филиал (614070, Россия, Пермь, ул. Студенческая, 38, e-mail: aimeyer@edu.hse.ru).

About the authors

Artem I. Meyer (Perm, Russian Federation) – Master’s Student, National Research University Higher School of Economics, Perm Branch (38, Studencheskaya st., Perm, 614070, e-mail: aimeyer@edu.hse.ru).

Библиографическое описание статьи согласно ГОСТ Р 7.0.100–2018:

Мейер, А.И. Разработка концептуальной архитектуры универсальной много-модульной платформы автономного портативного транспортного средства / А.И. Мейер. – DOI 10.15593/2499-9873/2020.4.06. – Текст: непосредственный // Прикладная математика и вопросы управления = Applied Mathematics and Control Sciences. – 2020. – № 4. – С. 89–102.

Цитирование статьи в изданиях РИНЦ:

Мейер А.И. Разработка концептуальной архитектуры универсальной много-модульной платформы автономного портативного транспортного средства // Прикладная математика и вопросы управления. – 2020. – № 4. – С. 89–102. DOI: 10.15593/2499-9873/2020.4.06

Цитирование статьи в references и международных изданиях:

Cite this article as:

Meyer A.I. The development of conceptual architecture of self-driving portable vehicle’s universal multi-modular platform. *Applied Mathematics and Control Sciences*, 2020, no. 4, pp. 89-102. DOI: 10.15593/2499-9873/2020.4.06 (in Russian)