

Обзор аппаратно-программного обеспечения систем управления роботов различного масштаба и назначения. Часть 2. Сервисная робототехника

А.М. Романов

МИРЭА – Российский технологический университет, Москва 119454, Россия

@Автор для переписки, e-mail: romanov@mirea.ru

В работе представлен обзор роботов различного масштаба и назначения. В ходе обзора анализируются применяемые аппаратные и программные решения и обобщаются наиболее распространенные структурные схемы систем управления. По результатам обзора проводится анализ подходов к масштабированию систем управления, применению алгоритмов интеллектуального управления, обеспечению отказоустойчивости, снижению массогабаритных размеров элементов систем управления, свойственных для разных классов роботов. Целью работы является поиск общих подходов, применяемых в различных областях робототехники для построения на их основе единой методологии проектирования масштабируемых интеллектуальных систем управления робототехническими комплексами с заданным уровнем отказоустойчивости на унифицированной элементной базе. Данная часть посвящена сервисной робототехнике. По результатам обзора сделаны следующие выводы: ключевой технологией в сервисной робототехнике с точки зрения масштабирования является Robot Operating System (ROS); сервисная робототехника является на сегодняшний день основным плацдармом для отработки интеллектуальных алгоритмов управления для тактического и стратегического уровня, которые интегрируются в общую систему на основе ROS; проблеме обеспечения отказоустойчивости в сервисной робототехнике практически не уделяется внимания, за исключением вопроса повышения надежности за счет изменения поведенческих алгоритмов; в ряде областей сервисной робототехники, в которых особо важно снижение массогабаритных размеров всех элементов, системы управления роботами выполняются на единственном вычислительном устройстве, в остальных случаях применяется многоуровневая архитектура, аналогичная тому, как это делается в промышленной робототехнике, с той разницей, что вместо программируемых логических контроллеров, на тактическом и стратегическом уровне используются встраиваемые компьютеры под управлением ROS и Linux.

Ключевые слова: робототехника, сервисные роботы, системы управления, мобильные роботы, дроны, беспилотные летательные аппараты, беспилотный транспорт, модульные реконфигурируемые роботы.

Для цитирования: Романов А.М. Обзор аппаратно-программного обеспечения систем управления роботов различного масштаба и назначения. Часть 2. Сервисная робототехника. *Российский технологический журнал*. 2019;7(6):68-86. <https://doi.org/10.32362/2500-316X-2019-7-6-68-86>

A review on control systems hardware and software for robots of various scale and purpose. Part 2. Service robotics

Aleksey M. Romanov

MIREA – Russian Technological University, Moscow 119454, Russia
@Corresponding author, e-mail: romanov@mirea.ru

A review of robotic systems was carried out. The paper analyzes applied hardware and software solutions and summarizes the most common block diagrams of control systems. The analysis of approaches to control systems scaling, the use of intelligent control, the achievement of fault tolerance, and the reduction of the weight and size of control system elements belonging to various classes of robotic systems were carried out. The goal of the review is finding common approaches used in various areas of robotics to build on their basis a uniform methodology for designing scalable intelligent control systems for robots with a given level of fault tolerance on a unified component base. This part is dedicated to service robotics. The following conclusions are made on the basis of the review results: the key technology in service robotics from the point of view of scalability is the Robot Operating System (ROS); service robotics is today the main springboard for testing intelligent algorithms for the tactical and strategic control levels that are integrated into a common system based on ROS; the problem of ensuring fault tolerance in the service robotics is practically neglected, with the exception of the issue of increasing reliability by changing behavioral algorithms; in a number of areas of service robotics, in which the reduction of mass and dimensions is especially important, the robot control systems are implemented on a single computing device, in other cases a multi-level architecture implemented on Linux-based embedded computers with ROS are used.

Keywords: robotics, service robots, control systems, mobile robots, drones, unmanned aerial vehicle, unmanned transport, modular reconfigurable robots.

For citation: Romanov A.M. A review on control systems hardware and software for robots of various scale and purpose. Part 2. Service robotics. *Rossiiskii tekhnologicheskii zhurnal* = Russian Technological Journal. 2019;7(6):68-86 (in Russ.). <https://doi.org/10.32362/2500-316X-2019-7-6-68-86>

Введение

Целью данного цикла работ является обзор и анализ существующих и перспективных робототехнических систем различного масштаба и назначения с точки зрения обобщения основных тенденций в области проектирования роботов и требований, предъявляемых к их аппаратному и программному обеспечению.

На основе анализа широкого спектра работ [1–5], так и или иначе затрагивающих вопросы классификации в робототехнике, выделено три наиболее крупных класса роботов по области применения: промышленная, сервисная и экстремальная робототехника. Данная часть посвящена обзору и анализу решений в области сервисной робототехники.

Сервисные роботы

Согласно стандарту ГОСТ Р ИСО 8373-2014 [6], сервисный (обслуживающий) робот – это робот, выполняющий полезную работу для людей и оборудования, за исключением промышленных задач по автоматизации. Таким образом, под этот термин попадает су-

шественное количество роботов, предназначенных для различных областей применений: бытовые роботы, научные роботы, автономные мобильные роботы, включая беспилотные транспортные средства, и многие другие. Тем не менее, несмотря на все многообразие сервисной робототехники, в ней, так же как и в промышленной, есть применяемые большинством разработчиков устоявшиеся инженерно-технические решения, описанию которых посвящен этот раздел.

Ключевой технологией в этой области робототехники на сегодня является Robot Operating System (ROS) [7–9]. ROS – это набор программных средств, исполняемых на базе операционной системы Linux, который предназначен для быстрого создания систем управления роботами.

В основе идеи ROS лежит модульный принцип создания программного обеспечения для робота. Каждый законченный алгоритм, составляющий закон управления, выполняется как отдельный процесс (Node в терминах ROS), который может обмениваться данными с другими процессами как циклически – через специальную систему подписок, так и по запросу – через создание сервиса [7] (рис. 1). Таким образом, созданный в одном проекте блок обработки камеры может быть использован без изменений для других роботов, достаточно лишь подписать остальные ROS Node на соответствующие циклически передаваемые данные (Topic в терминах ROS), содержащие целиком изображение или выделенные на его основе метаданные. Также, при помощи ROS можно абстрагировать, например, модуль планирования пути на местности от конкретных сенсоров, предоставляющих информацию о положении робота и препятствий, передав необходимые данные при помощи ROS Topic от соответствующих модулей или вообще подставив вместо показаний реального датчика данные из среды моделирования. По сути, реализация информационного обмена между приложениями и является ключевой функцией ROS.



Рис. 1. Пример структуры системы управления мобильного сервисного робота на базе ROS.

Помимо системы информационного обмена, в состав ROS входят подсистема 3D-визуализации, подсистема ведения журналов данных и событий, система визуализации графиков, средства для автоматизированной сборки, отладки и запуска приложений, входящих в состав системы управления роботом. Будучи созданным специально для сервисной робототехники [7], на сегодняшний день ROS применяется и в промышленных роботах. Для этих целей существует основанный на ROS проект ROS Industrial [10–12], который поддерживается консорциумом ведущих производителей в области промышленной автоматизации, таких как *Siemens*, *Bosch*, *ABB*, *Yaskawa*, *Comau* и др.

Бытовые сервисные роботы

Ярким примером бытового робота, основанного на ROS, является Willow Garage PR2 (рис. 2) [13]. Фактически система ROS впервые создавалась именно для него, а уже в последующем получила всемирное распространение.

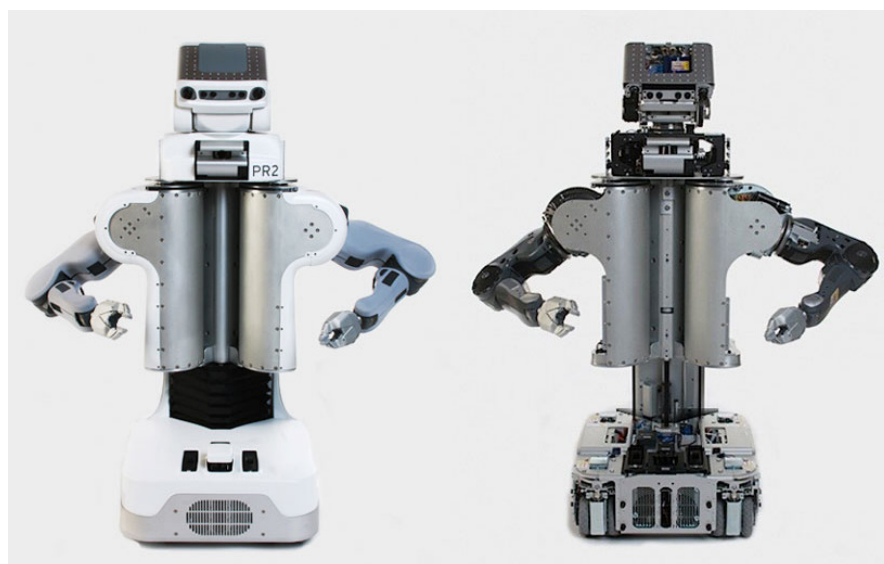


Рис. 2. Бытовой робот Willow Garage PR2.

Робот PR2 построен на модульном принципе, как механически, так и электрически, и представляет собой мобильную платформу, на которой установлены сменные модульные манипуляторы и опорно-поворотное устройство с сенсорами (рис. 3) [14].

Исполнительный уровень системы управления роботом реализован при помощи специализированных плат управления двигателями, построенных на базе программируемых логических интегральных схем (ПЛИС) [15]. Каждая такая плата обеспечивает замкнутый контур управления одним двигателем при помощи пропорционально-интегральных (ПИ) регуляторов, работающих с частотой 100 кГц. Платы, предназначенные для управления двигателями схватов, также отвечают за обработку тактильных датчиков, установленных на схватах. Подключение плат управления двигателями к тактическому уровню осуществляется по интерфейсу EtherCAT.

Структурная схема системы управления робота PR2 представлена на рис. 4.

Для реализации тактического и стратегического уровней используются два встраиваемых компьютера, каждый из которых построен на базе двух процессоров Intel Quad Core Xeон и оснащен 24 Гб оперативной памяти. Важно отметить, что данные компьютеры

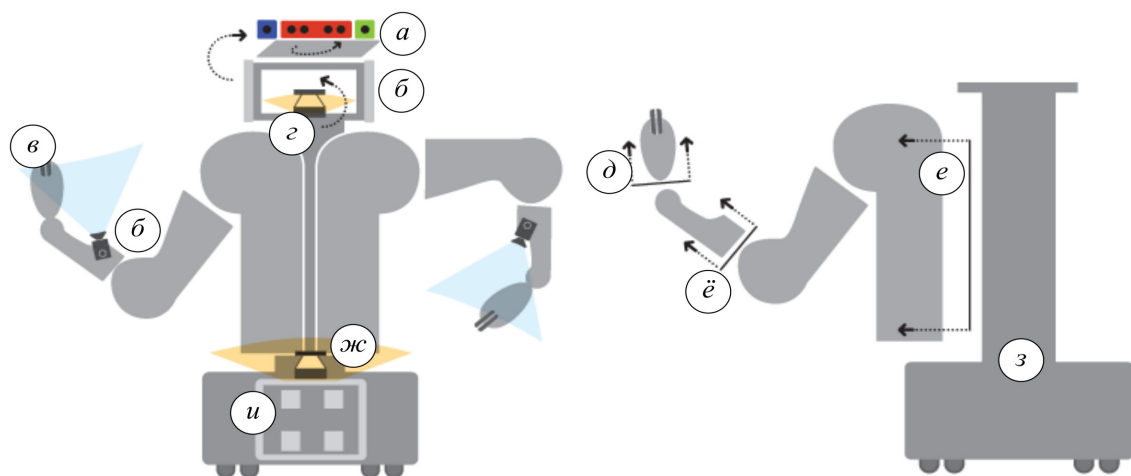


Рис. 3. Основные модули робота Willow Garage PR2: а) наклонно-поворотные камеры головы; б) боковая платформа для крепления дополнительных сенсоров; в) съемные датчики усилий на пальцах; г) наклонная платформа с лазерным дальномером; д) съёмные захватные устройства; е) съемные манипуляторы рук; ё) съемные манипуляторы предплечий; ж) лазерный дальномер; з) мобильная платформа; и) вычислительный модуль на базе двух компьютеров.

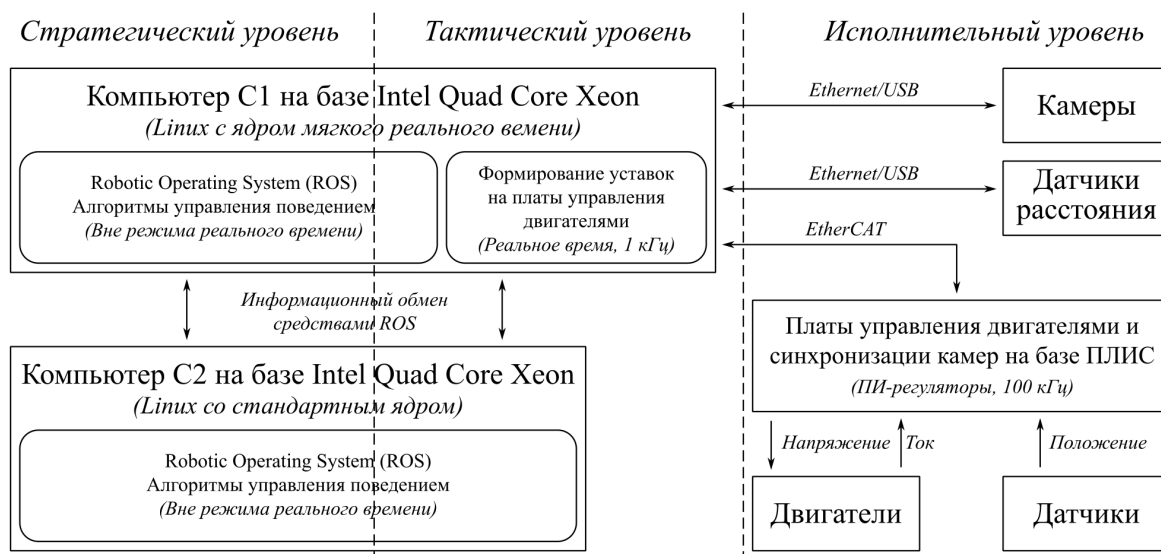


Рис. 4. Структурная схема системы управления робота Willow Garage PR2.

не являются взаимозаменяемыми и не служат для резервирования друг друга в случае аппаратных сбоев. Компьютер C1 (рис. 4) работает под управлением операционной системы Linux с патчем ядра, обеспечивающим работу в режиме мягкого реального времени [15]. На нем реализуется тактический уровень системы управления, производится формирование установок на платы управления двигателями и осуществляется информационный обмен по шине EtherCAT с частотой 1 кГц. Также он обеспечивает загрузку операционной системы на компьютер C2 и коммуникацию между всем компонентами робота. Компьютер C2 работает под управлением операционной системы Linux и служит для выполнения вспомогательных задач, в качестве дополнительных вычислительных

ресурсов для компьютера С1. Связь между программным обеспечением на компьютерах С1 и С2 осуществляется при помощи средств информационного обмена ROS. Для реализации алгоритмов управления стратегического уровня может использоваться любой из компьютеров С1, С2 или сразу оба, в то время как на тактическом уровне компьютер С2 используется только в качестве вспомогательного вычислителя для задач, не требующих реального времени. Подключение камер и датчиков расстояния осуществляется к компьютеру С1 по интерфейсам USB или Ethernet.

Мобильные сервисные роботы

Если оценивать общее количество проектов, то, наверное, одной из самых крупных категорий сервисных роботов являются мобильные платформы для всевозможных непромышленных применений (рис. 5): роботы для работы в медицинских учреждениях [16–18], беспилотные автомобили [19–24], научные и учебные роботы различного размера [25–33] и др. Ввиду своей многочисленности, для данной группы роботов характерно большое разнообразие применяемых аппаратно-программных средств. В то же время даже среди них можно выделить основные тенденции.

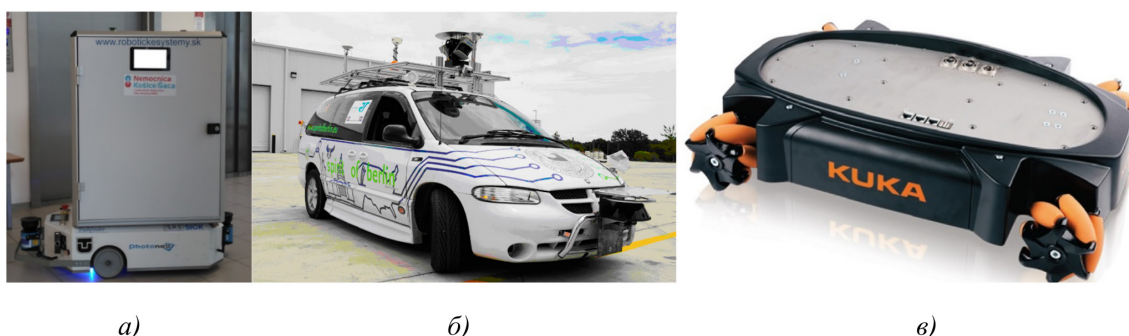


Рис. 5. Сервисные мобильные роботы различного типа: а) медицинский робот Pathfinder; б) беспилотный автомобиль Spirit of Berlin; в) научный робот KUKA youBot.

Так же, как и в промышленной, в мобильной сервисной робототехнике разработчики, в основном, предлагают использовать многоуровневые иерархические системы управления со структурой, аналогичной представленной на рис. 4, и явно выраженным исполнительным уровнем. В отдельных случаях встречаются решения, где все уровни управления реализованы на базе одного микроконтроллера [26, 29, 32], однако они характерны либо для относительно старых работ, либо в тех случаях, когда разработчики пытались достигнуть минимальной стоимости робота, преднамеренно отказываясь от возможности его дальнейшего масштабирования. В качестве платформы для реализации исполнительного уровня в мобильных сервисных роботах применяется весь набор существующих аппаратных средств – от промышленных систем управления на базе ПЛК и промышленных компьютеров [18] до специализированных плат на основе микроконтроллеров [20, 25, 33]. Среди учебных и экспериментальных роботов можно отметить популярность платформы Arduino как средства управления исполнительными устройствами [28, 29, 31].

Интерфейсы информационного обмена между исполнительным и тактическим уровнем в мобильных сервисных роботах можно разделить глобально на две большие группы: промышленные полевые шины (Ethernet POWERLINK, EtherCAT, CAN и т. д.) и меж-

платные интерфейсы связи (UART, I2C, SPI и т. д.). Первые чаще встречаются в роботах, предполагающих дальнейшее развитие в качестве коммерческого продукта [18, 19, 22], а вторые – в учебных и научно-исследовательских роботах [31, 33].

Наиболее общей чертой, присущей подавляющему большинству мобильных сервисных роботов, является использование ROS в качестве программной платформы. В некоторых случаях вместо ROS применяют программный пакет OROCOS [34–36], который также является инструментом создания модульного программного обеспечения, однако он имеет дополнительный функционал для поддержки работы в режиме реального времени. OROCOS появился в декабре 2000 г. и был популярен до появления ROS. На сегодняшний день ROS развивается и применяется значительно более активно, чем OROCOS, однако существуют работы, посвященные успешной интеграции обеих систем в рамках одного робота [16, 37, 38].

В качестве аппаратной платформы для запуска программного обеспечения в мобильных сервисных роботах используются встраиваемые компьютеры. Для малоразмерных роботов наибольшей популярностью пользуется Raspberry Pi различных модификаций [39]. В более крупных роботах, например, беспилотном транспорте [19, 20, 22], используются встраиваемые и промышленные компьютеры, а также серверы, преимущественно с архитектурой x86. В отдельных случаях для повышения производительности в рамках одного робота может использоваться несколько компьютеров [19, 22], аналогично тому, как это реализовано в роботе Willow Garage PR2 [15]. Наиболее сложные датчики, такие как видеокамеры и лазерные дальномеры, в большинстве случаев подключаются непосредственно к компьютеру тактического уровня, аналогично роботу PR2.

Стратегический уровень системы управления мобильных сервисных роботов реализуется на тех же вычислительных мощностях, что и тактический, или выносится на удаленный компьютер, преимущественно с x86-архитектурой [33, 40].

Сервисные роботы на элементной базе для любительских радиомоделей

Еще одной обособленной категорией с точки зрения аппаратного обеспечения являются сервисные роботы, построенные на элементной базе для любительских радиомоделей. К этой группе относятся практически все мультикоптеры, однако ими она не ограничивается. Высокие технические характеристики, малый вес и низкая стоимость современных компонентов для радиоуправляемых моделей сделали возможным изготовление на их основе автономных роботов подводного [41], надводного [42], наземного [43] и воздушного [44] базирования, а также роботизированных систем управления направленными антеннами [45] (рис. 6).

Роботы, относящиеся к данной категории, имеют характерные только для нее аппаратно-программное обеспечение и интерфейсы информационного обмена, которые будут рассмотрены ниже на примере одной из самых распространенных среди исследователей открытых платформ для создания беспилотных аппаратов различного типа базирования ArduPilot [46, 47]. Структурная схема системы управления роботом, построенным на базе ArduPilot, представлена на рис. 7.

В элементной базе любительских радиоуправляемых моделей присутствуют два основных вида исполнительных устройств: ведущие двигатели и управляемые по положению сервомашинки. Первые, как правило, служат для перемещения робота в простран-

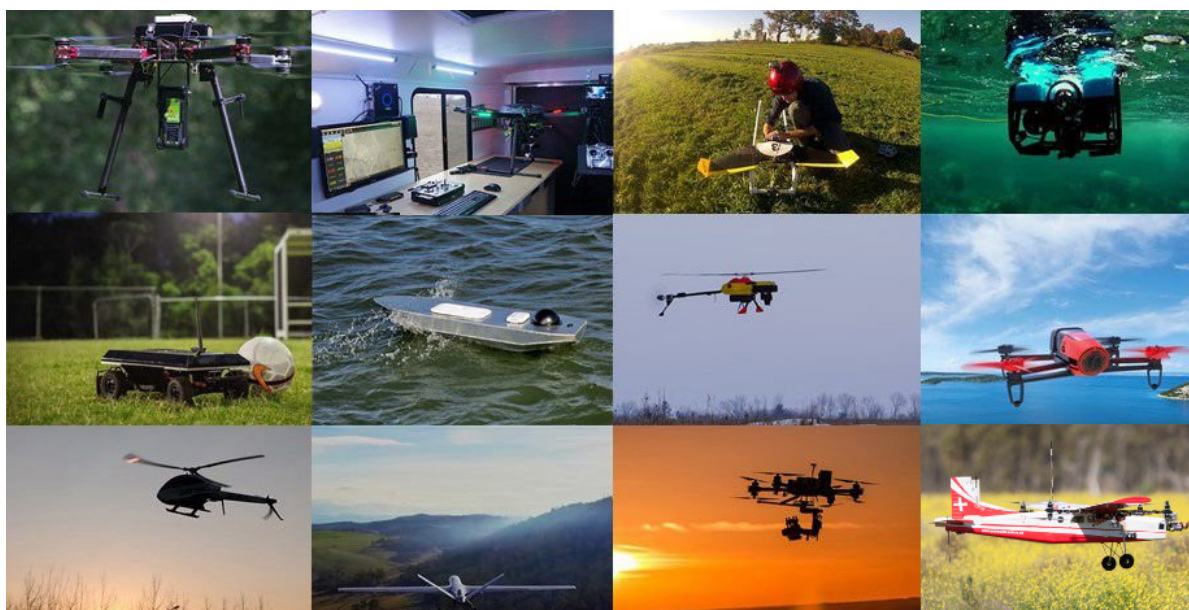


Рис. 6. Сервисные роботы, построенные на элементной базе для любительских радиоделей.

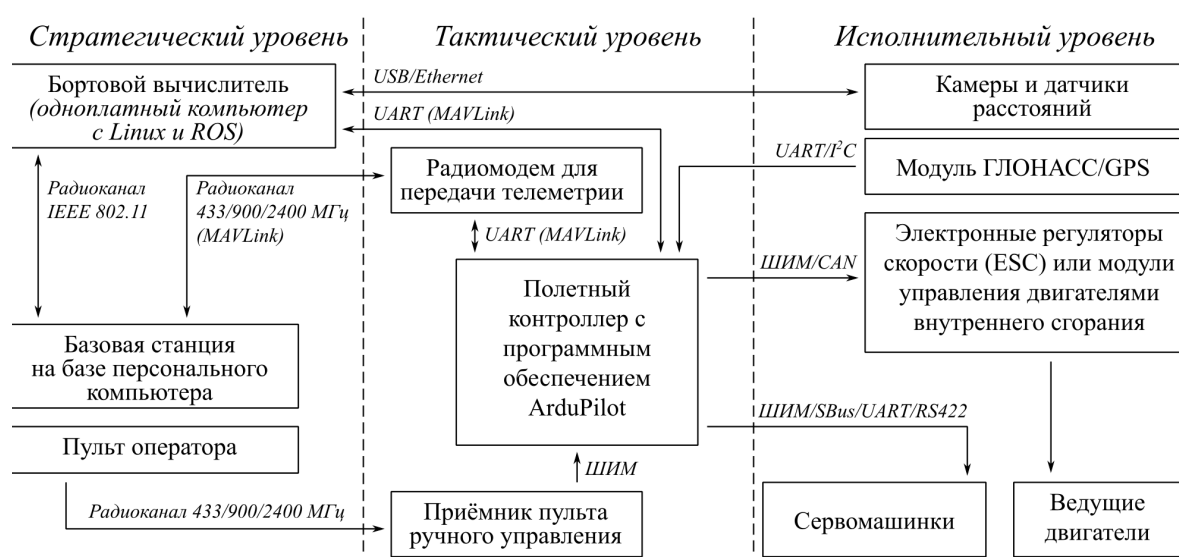


Рис. 7. Обобщенная структурная схема системы управления робота на базе ArduPilot.

стве, вращая колеса, гусеницы, воздушные, гребные винты или турбины, и представляют из себя электрический двигатель или двигатель внутреннего сгорания. Вторые используются для перемещения отдельных элементов робота друг относительно друга: поворота закрылков, колес, смещения центра тяжести, реализации подвесов для камеры или простейших манипуляторов.

В качестве ведущих двигателей на сегодняшний день наибольшей популярностью пользуются бесколлекторные двигатели постоянного тока (БДПТ). Мощность современных БДПТ для любительских радиоделей варьируется от единиц ватт до нескольких киловатт. Для управления такими двигателями используются электронные регуляторы скорости (ESC). Они осуществляют бездатчиковое управление и могут обеспечить

устойчивое вращение только на высоких скоростях, что приводит к определенным особенностям в поведении роботов, построенных на данной элементной базе. Некоторые из ESC могут управлять коллекторными двигателями постоянного тока (ДПТ). В качестве интерфейса управления ESC в подавляющем большинстве случаев используются широтно-импульсно-модулированные (ШИМ) дискретные сигналы, реже CAN. Для управления двигателями внутреннего сгорания, предназначенными для радиоделей, существуют специализированные модули, так же, как и ESC, управляемые при помощи ШИМ [46].

Радиомодельные сервомашинки представляют собой законченные устройства, включающие в себя двигатель, датчик, редуктор и систему управления, обеспечивающую перемещения оси сервомашинки на угол, задаваемый входной ШИМ. В некоторых сервомашинках вместо ШИМ используются цифровые каналы формирования установки: UART, RS422 и SBus [46]. В большинстве радиолюбительских сервомашин в качестве датчика применяется потенциометр. Система управления таких сервомашин, как правило, имеет единственный контур управления (по положению), реализованный при помощи пропорционально-интегрального (ПИ) регулятора.

Центральным элементом системы управления (рис. 7) является полетный контроллер. Данное название является устоявшимся и используется даже применительно к наземным и водным роботам [46]. Сейчас основным полетным контроллером для ArduPilot является Pixhawk [48] и его модификации, обладающие близкой элементной базой. Современные версии этого контроллера построены на 32 битном процессоре ARM Cortex-M4. Непосредственно на плате PixHawk установлена инерциальная навигационная система, включающая в себя барометр, магнитометры, гироскопы и акселерометры. Для подключения дополнительного оборудования он оснащен интерфейсами 5 UART, 1 I2C, 1 SPI и 2 CAN [46].

В качестве внешних информационно-измерительных устройств к полетному контроллеру, как правило, подключается модуль ГЛОНАСС/GPS и датчики расстояния, используемые для предотвращения столкновений. При полетах в помещениях вместо спутниковой навигации может быть использована локальная навигация по камере [48], радио- или ультразвуковым маякам [49]. Также к полетному контроллеру (рис. 7) могут быть подключены удаленный пульт оператора (ручного управления) и радиомодем для передачи телеметрии.

В иерархической модели управления [50] полетный контроллер решает задачи тактического уровня. Он обеспечивает требуемое положение и ориентацию мобильного робота в пространстве, управляя отдельными исполнительными устройствами, а также его перемещение к заданной точке. Для решения задач стратегического уровня в рассматриваемой категории роботов используют одноплатные компьютеры с операционной системой Linux и пакетом ROS, которые соединяются с полетным контроллером по протоколу MAVLink [48, 51, 52]. В мультиагентных системах канал MAVLink, как правило, ретранслируется на базовую станцию на базе персонального компьютера, где происходит дополнительная обработка данных и выдача управляющих команд для скоординированного управления всеми роботами в составе группы [53, 54].

Помимо ArduPilot, на сегодняшний день существует несколько аналогичных платформ [55], например, открытая Pixhawk PX4 и проприетарная DJI Matrice. Все они имеют схожую архитектуру. Ее главным недостатком является недостаточная масштаби-

руемость. Частично для того, чтобы обойти эту проблему, авторы таких проектов, как ArduPilot, используют в своем программном обеспечении отдельный слой абстракции от конкретного аппаратного обеспечения полетного контроллера. Однако даже с учетом данного слоя перенос программного обеспечения на другую микропроцессорную архитектуру или попытка повышения отказоустойчивости за счет внедрения дополнительного резервирования компонент потребует существенных трудозатрат. По сути, данные архитектуры оптимизированы для создания малоразмерных беспилотных аппаратов на элементной базе любительских радиомоделей и плохо подходят для создания роботов других типов.

Модульные реконфигурируемые мобильные роботы

Последняя категория сервисных роботов, которая заслуживает отдельного рассмотрения, это модульные реконфигурируемые роботы. На сегодняшний день такие роботы используются, в первую очередь, в научно-исследовательских организациях, где они являются объектом исследования, и мало применяются в других отраслях экономики. В то же время с точки зрения своего аппаратно-программного обеспечения они обладают уникальными характеристиками масштабирования и унификации используемой элементной базы. Также, среди всех рассмотренных ранее роботов, они предъявляют одни из самых жестких требований к габаритным размерам системы управления.

Модульные реконфигурируемые роботы (МРР) – это роботы, состоящие из большого количества унифицированных модулей, которые могут реорганизовываться в процессе функционирования в зависимости от решаемой задачи [56]. Под «реконфигурацией» и «реорганизацией» модулей для данной категории роботов подразумевается порядок их соединения между собой с целью изменения кинематической схемы. В отличие от других категорий сервисных роботов, МРР способны к масштабированию непосредственно в процессе работы, без отключения энергопитания робота, а повышенная отказоустойчивость в них достигается за счет того, что любой из модулей может потенциально заменить ранее вышедший из строя. По сути, речь идет о массовом многократном резервировании, в то время как даже в экстремальной робототехнике кратность резерва аппаратного обеспечения редко превышает значение 3.

На сегодняшний день различают три основных типа МРР с точки зрения конструкции модулей и способа их объединения: пространственно-решетчатые, цепные и мобильные [56, 57].

В пространственно-решетчатых МРР (рис. 8) модули располагаются в виде регулярного трехмерного шаблона, такого как, например, кубическая кристаллическая решетка. Эти системы используют строгую закономерность размещения модулей для облегчения вычислительных задач в ходе реконфигурации. К роботам данного класса относят ATRON [58], Odin [59], M-blocks [60].

Цепные МРР [56] характеризуются тем, что их модули объединяются в цепочки, из которых в дальнейшем строятся деревья или замкнутые контуры (рис. 9). К цепным роботам относятся PolyBot [61], MICROTUB [62], АРАКС [63, 64].

Также существуют гибридные МРР, такие как M-TRAN [65] (рис. 9 г), которые совмещают все принципы построения нескольких типов МРР, например, цепных пространственно-решетчатых.

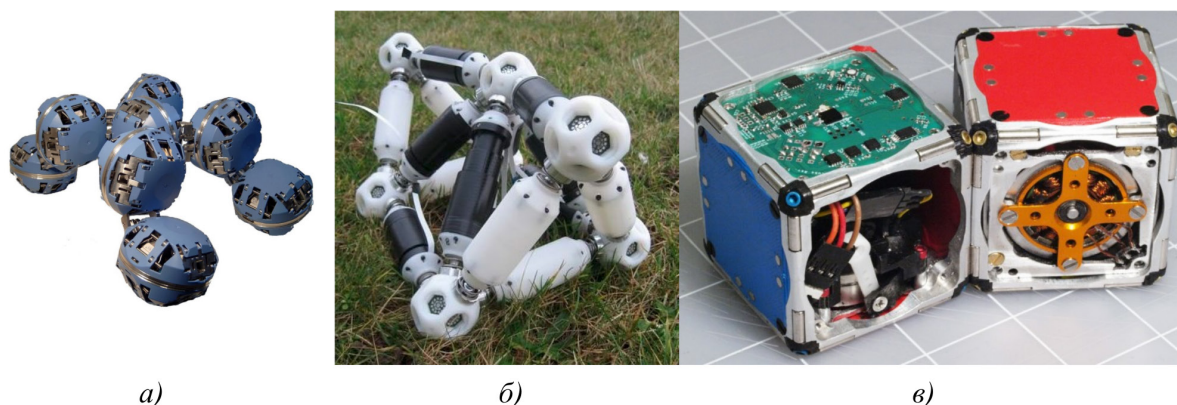


Рис. 8. Пространственно-решетчатые мобильные реконфигурируемые роботы:
а) ATRON; б) Odin; в) M Blocks.

Отличительной чертой мобильных МРР является возможность каждого отдельного модуля самостоятельно перемещаться в пространстве, объединяясь в различные структуры для выполнения конкретной задачи, сохраняя при этом возможность функционирования групп отдельных модулей в качестве роя [56]. К таким роботам относятся JL-I [66], M3Express [67], TEMP [68], MAPC [69] (рис. 10).

Также МРР принято разделять на гомогенные и гетерогенные [62, 70]. Гомогенными являются роботы, состоящие из полностью унифицированных модулей, такие как PolyBot [61], АРАКС [63], M-TRAN [65]. К гетерогенным МРР относят роботы, составленные из модулей разных типов, каждый из которых имеет определенную специализацию: блок управления, захватное устройство, источник питания и т. д. [62, 71]. На сегодняшний день гомогенные роботы, в первую очередь, служат объектом научных исследований, в то время как гетерогенные являются попыткой сделать архитектуру МРР более гибкой и оптимальной с точки зрения решения практических задач в сервисной и промышленной робототехнике [56, 57, 71, 72].

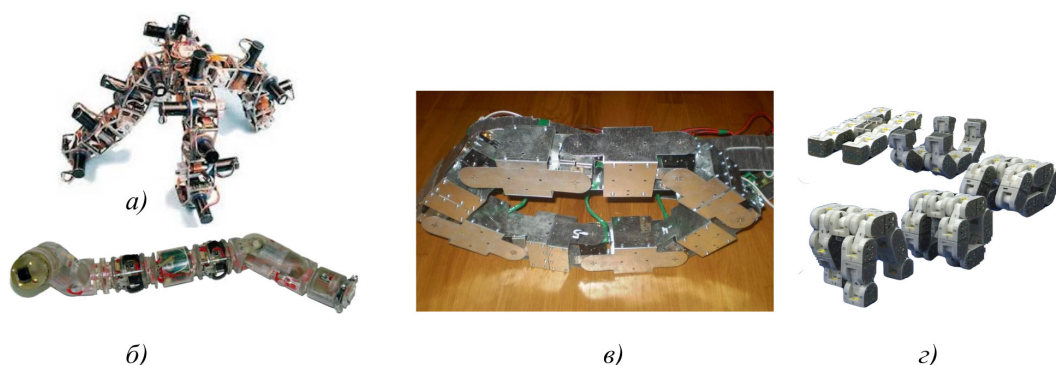


Рис. 9. Цепные и гибридные мобильные реконфигурируемые роботы:
а) PolyBot; б) MICROTUB; в) АРАКС; г) M-TRAN.

Для большинства МРР важнейшими характеристиками являются массогабаритные размеры модуля. Особенно критичны к этому параметру гомогенные МРР, так как их модули унифицированы и не обладают специализацией, а значит, каждый модуль должен содержать и привод, и источник питания, и вычислительное устройство. Вследствие этого во многих гомогенных МРР все уровни системы управления отдельного модуля

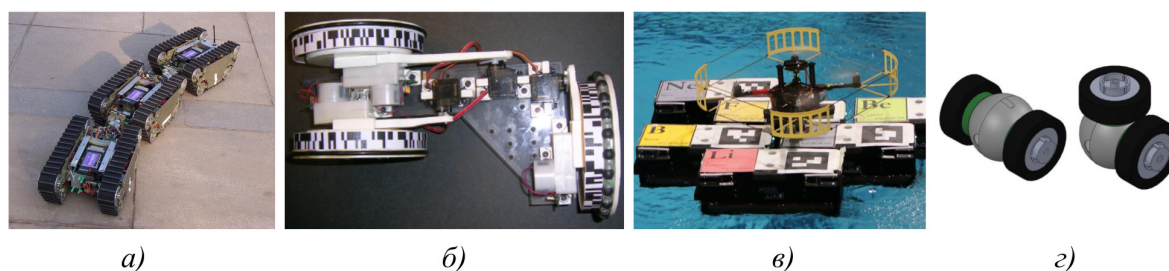


Рис. 10. Мобильные модульные реконфигурируемые роботы:
а) JL-I; б) M3Express; в) TEMP; г) MARC.

реализуются на базе одного вычислительного устройства, которое, как правило, представляет из себя микроконтроллер или сигнальный процессор [59, 61, 63, 67]. Гетерогенные роботы, а также крупные гомогенные роботы в качестве вычислительных устройств используют одноплатные компьютеры, на которых, как и в сервисных роботах других типов, может применяться ROS [68, 73]. Интересный подход с точки зрения создания программного обеспечения предлагают авторы [74, 75]. Он подразумевает использование для связи между отдельными компонентами библиотеки `zmq_robot` [76] и позволяет, с одной стороны, добиться модульности программных решений, схожей с ROS, а, с другой стороны, не требует операционной системы Linux, что позволяет использовать его на микроконтроллерах [75].

С точки зрения реализации алгоритмов управления MPP, на сегодняшний день существуют два основных подхода: централизованный и децентрализованный [77]. В централизованных системах предполагается наличие ведущего модуля, который осуществляет принятие решений и координирует все остальные модули в рамках общей задачи. Децентрализованный подход, наоборот, предполагает распределение задач между однотипными системами управления различных модулей. Системы с централизованной моделью управления проще в реализации и потенциально более эффективны с точки зрения достижения конечной цели, в то время как децентрализованные системы более отказоустойчивы и менее требовательны к вычислительным ресурсам, за счет распределения задач между различными модулями [78].

С точки зрения информационного обмена, централизованное управление, как правило, реализуется использованием глобальных схем связи между модулями, а децентрализованное – с использованием локальных. Однако из этого правила существуют исключения, например, [79]. В глобальных схемах связи применяются сети на базе CAN [61, 65] или Ethernet [75, 80], а также беспроводная связь IEEE802.15.1 [81, 82], IEEE802.15.4 [83, 84] и др. Беспроводные типы связи также используются для передачи целевой задачи роботу в целом, удаленного управления модулями и мониторинга их состояния. Глобальные схемы организации связи предполагают наличие уникального идентификатора у каждого модуля, по которому любой другой модуль может к нему обратиться.

В случае локальных схем межмодульной связи чаще всего используются пары приемник–передатчик, работающие по популярным протоколам RS-232, RS485, I2C, SPI. При таком подходе не требуется глобальная адресация модулей, однако стоит заметить, что и скорость обмена данными в системах с локальной связью существенно ниже, например, в модулях ATRON она не превышает 9600 б/с [58].

Существенным недостатком глобальных систем связи является невозможность автоматического определения взаимного расположения модулей робота. Для преодоления данного недостатка многие исследователи используют гибридные схемы связи, совмещающие в себе глобальный и локальный информационный обмен [78].

Интеллектуальное и групповое управление в сервисной робототехнике

С точки зрения внедрения технологий интеллектуального и группового управления, сервисные роботы являются наиболее представительным классом современных роботов. Это связано сразу с несколькими причинами:

1. К сервисным относят научно-исследовательских роботов, на которых отрабатываются передовые алгоритмы управления.
2. В отличие от промышленной робототехники, сервисные роботы часто работают в недетерминированной среде в условиях неопределенности.
3. Требования к надежности и повторяемости сервисных роботов в общем случае существенно ниже, чем у других категорий роботов, что позволяет оперативно внедрять в них новейшие решения, еще не прошедшие длительную апробацию.

Также стоит отметить, что подавляющее большинство рассмотренных работ, посвященных алгоритмам интеллектуального управления сервисными роботами, описывают решение траекторных и поведенческих задач, относящихся к стратегическому и тактическому уровням, в то время как на исполнительном уровне традиционно применяются линейные регуляторы. Основной платформой для реализации алгоритмов интеллектуального управления в сервисной робототехнике является ROS, для которого уже на данный момент существует обширная библиотека готовых решений [85]. Также для ускорения специфических вычислений, например, при использовании нейронных сетей, применяются специализированные ускорители, часто основанные на видеопроцессорах для графических карт [86–88].

На сервисных роботах отработаны как централизованные, так и децентрализованные методы группового управления [89–93]. Существуют готовые решения с открытым кодом для их реализации на базе наиболее крупных программных платформ, таких как ROS и ArduPilot [85, 94]. В качестве средства связи между роботами, входящими в состав единой мультиагентной системы, как правило, используются беспроводные сети стандартов IEEE 802.11, IEEE802.15.1 и IEEE802.15.4.

Основные тенденции в области создания систем управления сервисными роботами

Резюмируя проведенный анализ основных тенденций в области создания систем управления сервисных роботов, можно сделать следующие выводы:

- а) С точки зрения масштабирования, ключевой технологией в сервисной робототехнике является ROS. В качестве несомненных достоинств этой платформы можно выделить модульную программную архитектуру, унифицированные интерфейсы информационного обмена между компонентами, а также большее количество готовых модулей, созданных для различных областей прикладного применения сервисных роботов разработчиками со всего мира. Также для ROS существуют пакеты, предназначенные для ре-

шения задач группового управления как централизованными, так и децентрализованными методами. В то же время важно отметить, что ROS не обеспечивает работу в режиме реального времени, а также и то, что необходимым условием внедрения ROS является использование на вычислительном устройстве полноценной операционной системы Linux. Это существенным образом ограничивает применение ROS на исполнительном уровне систем управления сервисных роботов и, в конечном счете, приводит к отсутствию какой-либо унификации в этой области. Де факто для управления исполнительными устройствами в сервисной робототехнике используется весь спектр возможных аппаратных средств и методов организации информационного обмена, от макетных плат на маломощных микроконтроллерах, подключаемых по интерфейсам UART или SPI, до промышленных сервоприводов и систем ввода/вывода, подключаемых по полевым шинам реального времени.

б) Сервисная робототехника является на сегодняшний день основным плацдармом для отработки интеллектуальных алгоритмов управления для тактического и стратегического уровня. В этой области ежегодно публикуется большое количество работ. Основной платформой для интеграции этих алгоритмов в систему управления робота является ROS, в дополнение к которому в ряде случаев используются специализированные фреймворки и аппаратные ускорители, например, на базе видеопроцессоров для графических карт.

в) Обеспечению отказоустойчивости в сервисной робототехнике уделяется еще меньше внимания, чем в промышленной. В том числе, в отличие от промышленных, в сервисных роботах применяют отладочные платы и комплектующие, позиционируемые производителями для использования в любительской технике, которые имеют короткие назначенные сроки службы и часто для удешевления изготавливаются с использованием технологий пайки и сборки, не гарантирующих надежности при длительном функционировании. В то же время следует отметить, что в ряде работ, например, посвященных МРР, предлагаются методы преодоления отказов в работе, основанные не на увеличении надежности элементной базы, а на замещении функций неисправного модуля за счет изменения поведенческих алгоритмов оставшихся модулей.

г) Для ряда областей применения сервисной робототехники, таких как, например, беспилотные летательные аппараты и гомогенные МРР, снижение массогабаритных характеристик системы управления является ключевым требованием. Для таких применений создаются сервисные роботы, чьи системы управления построены на базе одного вычислительного устройства, реализующего все уровни управления. В то же время в остальной сервисной робототехнике исполнительный уровень обычно реализуется на отдельном аппаратном обеспечении аналогично тому, как это делается в промышленных роботах, а тактический и стратегический уровни – на базе одного или нескольких компьютеров под управлением Linux и ROS.

Литература / References:

1. Донченко А.А., Хрипунов С.В., Чиров Д.С. [и др.] Робототехнические средства, комплексы и системы военного назначения. Основные положения. Классификация. Методические рекомендации. М.: ФГБУ «ГНИИЦ РТ» МО РФ, 2015. 34 с.

[Donchenko A.A., Hripunov S.V., Chirov D.S. [et al.] Robotic equipment, complexes and military systems. The main provisions. Classification. Guidelines. Moscow: "GNIICR" RF MO (Main Research and Testing Center for Robotics of the Ministry of Defense of the Russian Federation), 2014. 34 p. (in Russ.).]

2. Юревич Е.И. Основы робототехники, 4 изд. СПб.: БХВ-Петербург, 2018. 304 с.
[Yurevich E.I. Basics of Robotics, 4th ed. SPb: BHV-Peterburg Publ., 2018. 304 p. (in Russ.).]
3. Zielinska T.T. History of service robots and new trends. In: Novel Design and Applications of Robotics Technologies. IGI Global, 2019:158-187. <https://doi.org/10.4018/978-1-5225-5276-5.ch006>
4. Kumar V., Bekey G., Zheng Y. Industrial, personal and service robots. In: Assessment of International Research and Development in Robotics. Ed. G. Bekey. World Technology Evaluation Center, Lancaster, 2006; pp. 41-48. <http://www.wtec.org/robotics/report/05-Industrial.pdf>
5. Лопота В.А., Юревич Е.И. Экстремальная робототехника и мехатроника. Принципы и перспективы развития. *Мехатроника, автоматизация, управление*. 2007;4:37-42.
[Lopota V.A., Yurevich E.I. Extreme robotics and mechatronics. Principles and perspectives of development. *Mekhatronika, avtomatizatsiya, upravlenie* = Mekhatronika, Avtomatizatsiya, Upravlenie. 2007;4:37-42 (in Russ.).]
6. ГОСТ Р ИСО 8373-2014. Роботы и робототехнические устройства. Термины и определения. М.: Стандартинформ, 2015. 16 с.
[GOST R ISO 8373-2012. Robots and robotic devices. Vocabulary. Moscow: Stahdartinform Publ., 2015. 16 p. (in Russ.).]
7. Quigley M., Conley K., Gerkey B.P., Faust J., Foote T., Leibs J., Wheeler, R., Ng A.Y. ROS: an open-source Robot Operating System. In: ICRA workshop on open source software. 2009;3(5): 6 p.
8. Garber L. Robot OS: A new day for robot design. *Computer*. 2013;46(12):16-20. <http://dx.doi.org/10.1109/MC.2013.434>
9. Koubâa A. (ed.). Robot Operating System (ROS). Verlag: Springer International Publishing, 2017. eBook ISBN 978-3-319-54927-9. <https://doi.org/10.1007/978-3-319-54927-9>
10. Zhang L., Merrifield R., Deguet A., Yang G-Z. Powering the world's robots-10 years of ROS. *Sci. Robotics*. 2017;2(11):eaar1868. <http://dx.doi.org/10.1126/scirobotics.aar1868>
11. Bouchier P. Embedded ROS [ROS topics]. *IEEE Robotics & Automation Magazine*. 2013;20(2):17-19. <http://dx.doi.org/10.1109/MRA.2013.2255491>
12. Romanov A., Romanov M., Slepynina E., Kholopov V. Analysis of ROS performance in terms of intelligent monitoring of discrete machinery manufacturing control systems. In: Proc. 15th Student Conference on Research and Development (SCORED). IEEE, 2017; pp. 13-17. <http://dx.doi.org/10.1109/SCORED.2017.8305429>
13. Bohren J., Rusu R.B., Marder-Eppstein E., Pantofaru C. Towards autonomous robotic butlers: Lessons learned with the PR2. In: Proc. 2011 IEEE International Conference on Robotics and Automation. IEEE, 2011; pp. 5568-5575. <https://doi.org/10.1109/ICRA.2011.5980058>
14. Willow Garage, PR2 Hardware Specs – [Electronic resource], URL:<http://www.willowgarage.com/pages/pr2/specs>
15. Willow Garage, PR2 User Manual, 2012 – [Electronic resource], URL:https://www.clearpathrobotics.com/wp-content/uploads/2014/08/pr2_manual_r321.pdf
16. Jakubiak J., Drwięga M., Stańczyk B. Control and perception system for ReMeDi robot mobile platform. In: Methods and Models in Automation and Robotics (MMAR). 2015 20th International Conference. IEEE, 2015; pp. 750-755. <https://doi.org/10.1109/MMAR.2015.7283969>
17. Arent K., Jakubiak J., Drwięga M., Cholewiński M. Control of mobile robot for remote medical examination: Design concepts and users' feedback from experimental studies. In: Proc. of the 9th International Conference on Human System Interactions (HSI 2016). IEEE, 2016; pp. 76-82. <http://dx.doi.org/10.1109/HSI.2016.7529612>
18. Bačík J., Durovsky F., Biroš M., Kyslan K. Pathfinder—development of automated guided vehicle for hospital logistics. *IEEE Access*. 2017;5(1):26892-26900. <https://doi.org/10.1109/ACCESS.2017.2767899>
19. Rojo J., Rojas R., Gunnarsson K., Simon M., Wiesel F., Ruf F., Wolter L., Zilly F., Santrac N., Ganjineh T., Sarkohi A., Ulbrich F., Latotzky D., Jankovic B., Hohl G., Wisspeintner T., May S., Pervoez K., Nowak W., Maurelli F., Droschel D. Spirit of Berlin: An Autonomous car for the DARPA urban challenge—Hardware and software architecture. Free Univ. Berlin, Germany, Tech. Rep., June 2007.
20. Gomez D., Marin P., Hussein A., de la Escalera A. ROS-based architecture for autonomous intelligent campus automobile (iCab). In: *UNED Plasencia Revista de Investigacion Universitaria*. Publisher: Agbatanero, 2016;12:257-272.
21. Shimchik I., Sagitov A., Afanasyev I., Matsuno F., Magid E. Golf cart prototype development and navigation simulation using ROS and Gazebo. MATEC Web of Conferences. 2016. V. 75. Article Number 09005. <https://doi.org/10.1051/mateconf/20167509005>
22. Jo K., Kim J., Kim D., Jang C. Development of autonomous car - Part II: A case study on the implementation of an autonomous driving system based on distributed architecture. *IEEE Trans. on Industrial Electronics*. 2015;62(8):5119-5132. <http://dx.doi.org/10.1109/TIE.2015.2410258>
23. Ferreira T., Garcia O., Vaqueiro J. Software Architecture for an Autonomous Car Simulation Using ROS, MORSE & A Qt Based Software for Control and Monitoring. In: XII Simpósio Brasileiro de automação Inteligente, 2015. 7 p.
24. Fernandes L.C., Souza J., Perrin G., Shinzato P.Y. CaRINA intelligent robotic car: architectural design and applications. *J. Systems Architecture*. 2014;60(4):372-392. <http://dx.doi.org/10.1016/j.sysarc.2013.12.003>
25. Лохин В.М., Манько С.В., Романов М.П., Гарцев И.Б., Трипольский П.Э., Александрова Р.И., Евстигнеев Д.В., Антипов О.А., Епишин С.В. Автономный мобильный мини-робот. *Известия Южного федерального*

университета. *Технические науки*. 2006;58(3):17-23.

[Lokhin V.M. [et al.] Autonomous mobile mini-robot. *Izvestiya SFedU. Engineering sciences*. 2006;58(3):17-23. (in Russ.)]

26. Doroftei I., Grosu V., Spinu V. Omnidirectional mobile robot-design and implementation. In book: *Bioinspiration and Robotics: Walking and Climbing Robots*. Ed. M.K. Habib. Vienna, Austria: I-Tech, 2007;511-528. <https://doi.org/10.5772/5518>

27. Bischoff R., Huggenberger U., Prassler E. KUKA youBot-a mobile manipulator for research and education. In: *Proc. IEEE International Conference on Robotics and Automation*. Shanghai, China, 2011:1-4. <http://dx.doi.org/10.1109/ICRA.2011.5980575>

28. Araújo A., Portugal D., Couceiro M.S., Rocha R.P. Integrating Arduino-based educational mobile robots in ROS. *Journal of Intelligent & Robotic Systems*. 2015;77(2):281-298. <https://doi.org/10.1007/s10846-013-0007-4>

29. López-Rodríguez F.M., Cuesta F. Andruino-A1: Low-cost educational mobile robot based on Android and Arduino. *Journal of Intelligent & Robotic Systems*. 2016;81(1):63-76. <http://dx.doi.org/10.1007/s10846-015-0227-x>

30. Jaskot K., Łakota T. Experimental mobile robot—Hardware. *Innovative Simulation Systems*. Springer, Cham, 2016;277-289.

31. Морозов А.А., Русаков Д.А., Казачек Н.А. Программно-аппаратный комплекс на базе мобильных роботов с изменяемой геометрией шасси // *Материалы VI Международ. научно-техн. конф. молодых ученых*, 2016. Омск: Изд-во Омский гос. технический ун-т. 2016. С. 146–150.

[Morozov A.A., Rusakov D.A., Kazachek N.A. Hardware-software complex based on mobile robots with variable chassis geometry. *Materials of the VI Int. Scientific and Technical Conf. of Young Scientists*, 2016. Omsk: Publishing house of the Omsk State Technical University, 2016;146-150 (in Russ.)]

32. Mondada F., Bonani M., Riedo E., Briod M. Bringing robotics to formal education: The thymio open-source hardware robot. *IEEE Robotics & Automation Magazine*. 2017;24(1): 77-85. <https://doi.org/10.1109/MRA.2016.2636372>

33. Alberri M., Hegazy S., Badra M., Nasr M. Generic ROS-based Architecture for Heterogeneous Multi-Autonomous Systems Development. In: *2018 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. IEEE, 2018. 6 p. <https://doi.org/10.1109/ICVES.2018.8519589>

34. Bruyninckx H. Open robot control software: the OROCOS project. In: *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation*. IEEE, 2001;3:2523-2528. <https://doi.org/10.1109/ROBOT.2001.933002>

35. Schlegel C. A component approach for robotics software: Communication patterns in the OROCOS context. In: *Autonome Mobile Systeme 2003*. Dillmann R., Wörn H., Gockel T. (eds). Berlin, Heidelberg: Springer, 2003; pp. 253-263. https://doi.org/10.1007/978-3-642-18986-9_26

36. Михайлова У.В., Михайлов Е.А., Сарваров А.С. Программные решения для разработки архитектуры системы управления роботом. *Электротехнические системы и комплексы*. 2013;(21):111-117.

[Mikhailova V.V., Mikhailov E.A., Sarvarov A.S. Software solutions for the development of robot control system architecture. *Elektrotehnicheskie sistemy i komplekсы* = Electrotechnical systems and complexes. 2013;(21):111-117 (in Russ.)]

37. Smits R., Bruyninckx H. Composition of complex robot applications via data flow integration. In: *Proceedings 2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011; pp. 5576-5580. <http://dx.doi.org/10.1109/ICRA.2011.5979958>

38. Buys K., Bellens S., Vanthienen N., Decre W., Klotzbucher M., De Laet T., Smits R., Bruyninckx H., De Schutter J. Haptic coupling with the PR2 as a demo of the OROCOS-ROS-Blender integration. In: *IROS PR2 Workshop*. San Francisco, California. 2011;25:30.

39. Петин В. Микрокомпьютеры Raspberry Pi: Практическое руководство. СПб.: БХВ-Петербург, 2015. 240 с. ISBN 978-5-9775-3519-9

[Petin V. *Microcomputers Raspberry Pi: A Practical Guide*. SPb.: BHV-Peterburg Publ., 2015. 240 p. ISBN 978-5-9775-3519-9 (in Russ.)]

40. Манько С.В., Диане С.А., Новосельский А.К. Макетный образец многоагентной робототехнической системы на базе платформы KUKA youBot. *Труды Международной научно-технической конференции «Экстремальная робототехника»* (Санкт-Петербург, 2015). СПб.: Политехника-сервис, 2015. С. 210–214.

[Manko S.V., Diane S.A., Novoselsky A.K. A prototype of multi-agent robotic system based on KUKA youBot platform. *Proceedings of the International Scientific and Technical Conference “Extreme Robotics”* (St. Petersburg, 2015). SPb.: Polytechnic service, 2015; pp. 210-214. (in Russ.)]

41. Schillaci G., Schillaci F., Hafner V.V. A Customisable underwater robot. *arXiv preprint arXiv:1707.06564*. 2017. <https://arxiv.org/abs/1707.06564>

42. Carlson D.F., Rysgaard S. Adapting open-source drone autopilots for real-time iceberg observations. *MethodsX*. 2018;5:1059-1072. <http://dx.doi.org/10.1016/j.mex.2018.09.003>

43. Simon G.A., Moore J.M., Clark A.J., McKinley P.K. Evo-ROS: integrating evolution and the robot operating system. In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. ACM, 2018; pp. 1386-1393. <http://dx.doi.org/10.1145/3205651.3208269>

44. Ebeid E., Skriver M., Terkildsen K.H., Jensen K. A survey of open-source UAV flight controllers and flight simulators. *Microprocessors and Microsystems*. 2018;61:11-20. <http://dx.doi.org/10.1016/j.micpro.2018.05.002>

45. Pop S., Luculescu M.C., Cristea L., Zamfira C.S., Boer A.L. Improving communication between unmanned

aerial vehicles and ground control station using antenna tracking systems. In: Auer M., Zutin D. (eds) Online Engineering & Internet of Things. Lecture Notes in Networks and Systems. V. 22. Springer, Cham, 2018; pp. 532-539. https://doi.org/10.1007/978-3-319-64352-6_49

46. ArduPilot Documentation – [Electronic resource]: URL:<http://ardupilot.org/ardupilot/index.html>

47. Журавлёв Д.О., Наинг З.Х. Эволюция систем управления беспилотных летательных аппаратов: от появления до наших дней. Достижения и перспективы современной науки. Материалы Междунар. научно-практической конф. Нефтекамск: Научно-издательский центр «Мир науки», 2017. С. 57-87.

[Zhuravlev D.O., Naing Z.H. The evolution of control systems for unmanned aerial vehicles: from appearance to the present day. In: Achievements and Prospects of Modern Science. Proceed. of the International Scientific and Technical Conference. Neftkamsk: Scientific Publishing Center “Mir nauki”, 2017. P. 57-87 (in Russ.).]

48. Meier L., Tanskanen P., Heng L., Lee G.H. PIXHAWK: A micro aerial vehicle design for autonomous flight using onboard computer vision. *Autonomous Robots*. 2012;33(1-2):21-39. <http://dx.doi.org/10.1007/s10514-012-9281-4>

49. Li Y., Scanavino M., Capello E., Dabbene F. A novel distributed architecture for UAV indoor navigation. *Transportation Research Procedia*. 2018;35:13-22. <http://dx.doi.org/10.1016/j.trpro.2018.12.003>

50. Макаров И.М., Лохин В.М. Интеллектуальные системы автоматического управления. М.: Физматлит, 2001. 576 с.

[Makarov I.M., Lokhin V.M. Intelligent automatic control systems. Moscow: Fizmatlit Publ., 2001. 576 p. (in Russ.).]

51. Abeywardena D., Pounds P., Dissanayake G. Design and development of ReCOPTER: An open source ROS-based multi-rotor platform for research. In: Proceed. Austral. Conf. Robot. Autom. (ACRA). 2015. 10 p. <http://hdl.handle.net/10453/117849>

52. Meier L. Dynamic Robot Architecture for Robust Realtime Computer Vision: Doctoral Thesis. ETH Zürich, 2017. <https://doi.org/10.3929/ethz-a-010874068>

53. Koubaa A., Qureshi B. DroneTrack: Cloud-based real-time object tracking using unmanned aerial vehicles over the Internet. *IEEE Access*. 2018;6:13810-13824. <https://doi.org/10.1109/ACCESS.2018.2811762>

54. Lamping A.P., Ouwerkerk J.N., Cohen K. Multi-UAV Control and Supervision with ROS. In: Proceed. 2018 Aviation Technology, Integration, and Operations Conference. At Atlanta, GA, Volume: AIAA AVIATION Forum. 2018; p. 4245. <https://doi.org/10.2514/6.2018-4245>

55. Chen S., F Laefer D., Mangina E. State of technology review of civilian UAVs. *Recent Patents on Engineering*. 2016;10(3):160-174. <https://doi.org/10.2174/1872212110666160712230039>

56. Seo J., Paik J., Yim M. Modular reconfigurable robotics. *Annual Review of Control, Robotics, and Autonomous Systems*. 2019;2:63-88. <https://doi.org/10.1146/annurev-control-053018-023834>

57. Alattas R.J., Patel S., Sobh T. M. Evolutionary modular robotics: Survey and analysis. *Journal of Intelligent & Robotic Systems*. 2019;95(3-4):815-828. <https://doi.org/10.1007/s10846-018-0902-9>

58. Ostergaard E.H., Kassow K., Beck R., Lund H.H. Design of the ATRON lattice-based self-reconfigurable robot. *Autonomous Robots*. 2006;21(2):165-183. <http://dx.doi.org/10.1007/s10514-006-8546-1>

59. Garcia R.F.M., Lyder A., Christensen D.J., Stoy K. Reusable electronics and adaptable communication as implemented in the odin modular robot. In: Proceed. IEEE International Conference on Robotics and Automation. IEEE, 2009; pp. 1152-1158. <http://dx.doi.org/10.1109/ROBOT.2009.5152811>

60. Romanishin J. W., Gilpin K., Rus D. M-blocks: Momentum-driven, magnetic modular robots. In: Proc. 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2013; pp. 4288-4295. <https://doi.org/10.1109/IROS.2013.6696971>

61. Yim M., Duff D.G., Roufas K.D. PolyBot: a modular reconfigurable robot. In: Proceed. of the IEEE International Conference on Robotics and Automation (ICRA'00). San Francisco, Calif, USA, 2000. V. 1. P. 514-520. <https://doi.org/10.1109/ROBOT.2000.844106>

62. Brunete A., Hernandj M., Gambao E., Torres J.E. A behaviour-based control architecture for heterogeneous modular, multi-configurable, chained micro-robots. *Robotics and Autonomous Systems*. 2012;60(12):1607-1624. <https://doi.org/10.1016/j.robot.2012.09.019>

63. Макаров И.М., Лохин В.М. Технологии обработки знаний в задачах управления автономными мехатронно-модульными реконфигурируемыми роботами. *Информационные технологии*. 2010;(S8):1-32.

[Makarov I.M., Lokhin V.M. Knowledge processing technologies for autonomous mechatronic-modular reconfigurable robots control tasks. *Informatsionnye tekhnologii* (Information Technology). 2010;(S8):1-32 (in Russ.).]

64. Макаров И.М., Лохин В.М., Манько С.В., Романов М.П., Крюченков Е.Н., Кучерский Р.В., Диане С.А. Мультиагентные робототехнические системы: примеры и перспективы применения. *Мехатроника, автоматизация, управление*. 2012; (2):22-32.

[Makarov I.M. [et al.] Multi-agent robotic systems: examples and prospects of application. *Mekhatronika, avtomatizatsiya, upravlenie* [Mechatronics, Automation, Control]. 2012;(2):22-32 (in Russ.).]

65. Murata S., Kurokawa H. Self-organizing robots. Volume 77 of Springer tracts in advanced robotics. NY, USA: Springer, 2012. <http://dx.doi.org/10.1007/978-4-431-54055-7>

66. Guanghua Z., Zhicheng D., Wei W. Realization of a modular reconfigurable robot for rough terrain. In: Proceed. of the 2006 IEEE International Conference on Mechatronics and Automation. 2006; pp. 289-294. <http://dx.doi.org/10.1109/ICMA.2006.257529>

67. Wolfe K.C., Moses M., Kutzer M., Chirikjian G.S. M 3 Express: a low-cost independently-mobile reconfigurable modular robot. In: Proceed. of 2012 IEEE International Conference on Robotics and Automation. 2012; pp. 2704-2710. <http://dx.doi.org/10.1109/ICRA.2012.6224971>
68. O'Hara I., Paulos J., Davey J., Eckenstein N. Self-assembly of a swarm of autonomous boats into floating structures. In: Proceed. of the IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2014; pp. 1234-1240. <http://dx.doi.org/10.1109/ICRA.2014.6907011>
69. Павлюк Н.А., Крестовников К.Д., Пыхов Д.Э. Мобильная автономная реконфигурируемая система. *Проблемы региональной энергетики*. 2018;1(36):125-135. <http://dx.doi.org/10.5281/zenodo.1217296>
- [Pavlyuk N.A., Krestovnikov K.D., Pykhov D.E. Mobile autonomous reconfigurable system. *Problemy regionalnoi energetiki* = Problems of Regional Energetics. 2018;1(36):125-135 (in Russ.). <http://dx.doi.org/10.5281/zenodo.1217296>]
70. Андреев В.П., Ким В.Л., Подураев Ю.В. Сетевые решения в архитектуре гетерогенных модульных мобильных роботов. *Робототехника и техническая кибернетика*. 2016;3:23-29.
- [Andreev V.P., Kim V.L., Poduraev Yu.V. Network-based design of heterogeneous modular mobile robotic systems. *Robototekhnika i tekhnicheskaya kibernetika* = Robotics and Technical Cybernetics. 2016;3:23-29 (in Russ.).]
71. Андреев В.П., Подураев Ю.В. Функционально-модульный принцип построения гетерогенных мобильных роботов. Труды Международной научно-технической конференции «Экстремальная робототехника». СПб.: ООО «АП4Принт», 2016. С. 39–49.
- [Andreev V.P., Poduraev Yu.V. Functional-modular design of heterogeneous mobile robotic systems. In: Proceed. of the International Scientific and Technological Conference “Extreme Robotics”. Saint-Petersburg: AP4Print Publ., 2016. P. 39-49 (in Russ. / Engl.).]
72. Лопота А.В., Юревич Е.И. Этапы и перспективы развития модульного принципа построения робототехнических систем. *Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Информатика. Телекоммуникации. Управление*. 2013;1(164):98-103.
- [Lopota A.V., Yurevich E.I. Stages and development prospects of robotic systems design modular principle. *Nauchno tekhnicheskie ведомosti Sankt-Peterburgskogo gosudarstvennogo politekhnicheskogo universiteta. Informatika. Telekommunikatsii. Upravlenie* = St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems. 2013;1(164):98-103 (in Russ.).]
73. Thakker R., Kamat A., Bharambe S., Cheddarwar S.S., Bhurchandi K.M. Rebis-reconfigurable bipedal snake robot. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2014; pp. 309-314. <https://doi.org/10.1109/iros.2014.6942577>
74. Андреев В.П., Ким В.Л., Плетенев П.Ф. Программно-аппаратное решение оперативного реконфигурирования гетерогенных роботов. *Мехатроника, автоматизация, управление*. 2018;19(6):387-395. <http://novtex.ru/mech/eng/doi/mau.19.387-395.html>
- [Andreev V.P., Kim V.L., Pletenev P.F. Hardware and software solution for rapid reconfiguration of heterogeneous robots. *Mekhatronika, avtomatizatsiya, upravlenie* = Mechatronics, automation, control. 2018;19(6):387-395 (in Russ.). <http://novtex.ru/mech/eng/doi/mau.19.387-395.html>]
75. Андреев В.П., Плетенев П.Ф. Метод информационного взаимодействия для систем распределенного управления в роботах с модульной архитектурой. *Труды СПИИРАН*. 2018;2(57):134-160. <https://doi.org/10.15622/sp.57.6>
- [Andreev V.P., Pletenev P.F. Method of information interaction for distributed control systems of robots with modular architecture. *Trudy SPIIRAN* = SPIIRAS Proceedings. 2018;2(57):134-160 (in Russ.). <https://doi.org/10.15622/sp.57.6>]
76. Kirsanov K. Software architecture of control system for heterogeneous group of mobile robots. *Procedia Engineering*. 2015;100:278-282. <https://doi.org/10.1016/j.proeng.2015.01.368>
77. Brunete A., Ranganath A., Segovia S., de Frutos J.P., Hernando M., Gambao E. Current trends in reconfigurable modular robots design. *International Journal of Advanced Robotic Systems*. 2017;14(3):1-21. <https://doi.org/10.1177/1729881417710457>
78. Yim M., White P., Park M., Sastra J. Modular self-reconfigurable robots. In: Encyclopedia of complexity and systems science / eds. R.A. Meyers. New York: Springer, 2009. P. 5618-5631. https://doi.org/10.1007/978-0-387-30440-3_334
79. Zhang Y., Roufas K. D., Yim M. Software architecture for modular self-reconfigurable robots. *Intelligent Robots and Systems*, 2001. Proceed. of the 2001 IEEE/RSJ International Conference on. IEEE, 2001. V. 4; pp. 2355-2360.
80. Liedke J., Matthias R., Winkler L., Wörn H. The collective self-reconfigurable modular organism (CoSMO). In: 2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics. IEEE, 2013; pp. 1-6. <https://doi.org/10.1109/AIM.2013.6584059>
81. Moeckel R., Jaquier C., Drapel K., Dittrich E. Exploring adaptive locomotion with YaMoR, a novel autonomous modular robot with Bluetooth interface. *Industrial Robot: An International Journal*. 2006;33(4):285-290. <https://doi.org/10.1108/01439910610667908>
82. Zykov V., Chan A., Lipson H. Molecubes: An open-source modular robotics kit. In: IROS-2007 Self-Reconfigurable Robotics Workshop. 2007; pp. 3-6.
83. Qiao G., Song G., Zhang J., Sun H., Wang W., Song A. Design of transmute: a modular self-reconfigurable robot with versatile transformation capabilities. In: 2012 IEEE International Conference on Robotics and Biomimetics (ROBIO). IEEE, 2012; pp. 1331-1336. <https://doi.org/10.1109/ROBIO.2012.6491153>
84. Davey J., Kwok N., Yim M. Emulating self-reconfigurable robots-design of the SMORES system. In: 2012 IEEE/

RSJ International Conference on Intelligent Robots and Systems. IEEE, 2012; pp. 4464-4469. <https://doi.org/10.1109/IROS.2012.6385845>

85. ROS Index: Packages – [Electronic resource]: URL:<https://index.ros.org/packages/>

86. Buyval A., Afanasyev I., Magid E. Comparative analysis of ROS-based Monocular SLAM methods for indoor navigation. In: 9th International Conference on Machine Vision (ICMV 2016). International Society for Optics and Photonics, 2017. V. 10341. P. 103411K. <http://dx.doi.org/10.1117/12.2268809>

87. Lentin J. ROS Robotics Projects. Packt Publ. Ltd, 2017. 452 p.

88. Zhou C., Li F., Cao W., Wang C. Design and implementation of a novel obstacle avoidance scheme based on combination of CNN-based deep learning method and liDAR-based image processing approach. *Journal of Intelligent & Fuzzy Systems*. 2018;35(3):1-11. <https://doi.org/10.3233/JIFS-169706>

89. Hennes D., Claes D., Meeussen W., Tuyls K. Multi-robot collision avoidance with localization uncertainty. In: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems. Vol. 1. International Foundation for Autonomous Agents and Multiagent Systems, 2012 147-154.

90. Reid R., Cfn A., Meiklejohn C., Boeing A., Braunl T. Cooperative multi-robot navigation, exploration, mapping and object detection with ROS. In: 2013 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2013;1083-1088. <https://doi.org/10.1109/IVS.2013.6629610>

91. Rubenstein M., Ahler C., Hoff N., Cabrera A., Nagpal R. Kilobot: A low cost robot with scalable operations designed for collective behaviors. *Robotics and Autonomous Systems*. 2014;62(7):966-975. <https://doi.org/10.1016/j.robot.2013.08.006>

92. Lee B. H. Y., Morrison J. R., Sharma R. Multi-UAV control testbed for persistent UAV presence: ROS GPS waypoint tracking package and centralized task allocation capability. In: 2017 International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, 2017;1742-1750. <https://doi.org/10.1109/ICUAS.2017.7991424>

93. Siwek M., Besseghieur K., Baranowski L. The effects of the swarm configuration and the obstacles placement on control signals transmission delays in decentralized ROS-embedded group of mobile robots. AIP Conference Proceedings. AIP Publishing, 2018;2029(1):020069.

94. ArduPilot Swarming — Mission Planner documentation – [Electronic resource], URL:<http://ardupilot.org/planner/docs/swarming.html>

Об авторе:

Романов Алексей Михайлович, кандидат технических наук, доцент кафедры биокбернетических систем и технологий Института кибернетика ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр. Вернадского, д. 78).

About the author:

Alexey M. Romanov, Cand. of Sci. (Engineering), Associate Professor, Chair of Biocybernetics Systems and Technologies, Institute of Cybernetics, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow 119454, Russia).