



## ОБ ОДНОМ ПОДХОДЕ К ОРГАНИЗАЦИИ БЫСТРОГО ПРОТОТИПИРОВАНИЯ РОБОТОВ В УСЛОВИЯХ ЖЁСТКИХ ОГРАНИЧЕНИЙ

Карпов В. Э.<sup>1</sup>, доктор техн. наук, доцент, [Karpov\\_VE@nrcki.ru](mailto:Karpov_VE@nrcki.ru)

Сорокоумов П. С.<sup>1</sup>, научный сотрудник, [Sorokoumov\\_PS@nrcki.ru](mailto:Sorokoumov_PS@nrcki.ru)

Карпов В. В.<sup>1</sup>, инженер-исследователь, [Karpov\\_VV@nrcki.ru](mailto:Karpov_VV@nrcki.ru)

Воробьев В. В.<sup>1</sup>, зам. начальника лаборатории, ✉ [Vorobev\\_VV@nrcki.ru](mailto:Vorobev_VV@nrcki.ru)

<sup>1</sup>НИЦ Курчатовский институт, пл. Академика Курчатова, д. 1, 123182, Москва, Россия

### Аннотация

В работе описан подход к организации процесса быстрого прототипирования наземных групповых робототехнических комплексов малыми коллективами разработчиков в сжатые сроки. Прототипирование осуществляется путём последовательного построения моделей разного уровня абстракции. Показано, что основные затраты ресурсов при переходе между этими описаниями могут быть уменьшены за счет использования стандартных поведенческих моделей роботов и применения библиотек типовых программных конструкторских решений. Применение поведенческих моделей на основе механизма мета-автоматов позволяет, используя абстрактные реализации на математическом, имитационном и физическом уровнях, получать вполне практические и работоспособные решения. Базовым инструментальным средством разработки систем управления и проведения физического моделирования является Robot Operating System (ROS). Применение этого подхода продемонстрировано на примере транспортной задачи, решаемой роботами с простейшей сенсорикой без глобальной навигации.

**Ключевые слова:** *групповая робототехника, моделирование, поведенческие модели, проектный подход к обучению.*

**Цитирование:** Карпов В. Э., Сорокоумов П. С., Карпов В. В., Воробьев В. В. Об одном подходе к организации быстрого прототипирования роботов в условиях жёстких ограничений // Компьютерные инструменты в образовании. 2024. № 3. С. 79–92. doi:10.32603/2071-2340-2024-3-79-92

### 1. ВВЕДЕНИЕ

Процесс разработки робототехнических систем (РТС) от создания математической модели до проведения итоговых испытаний давно определен и стандартизован. В зависимости от сложности изделия можно заранее оценить материальные и временные затраты, определить состав коллектива, подобрать средства проектирования и т. д. Здесь сложно и вряд ли нужно что-то придумывать, однако существует некоторая специфическая область проектирования РТС, для которой процесс разработки осуществляется в условиях принципиальных ограничений. Речь идет об особом классе РТС, которые

должны быть созданы за строго определенное время, а состав коллектива разработчиков фиксирован. Кроме того, важной особенностью является то, что а) коллектив временный, создается на некоторый непродолжительный период (менее года) и б) коллектив нацелен на оперативное решение достаточно широкого спектра задач, основанных на едином программно-аппаратном базисе (в основном меняется функционал базовых конструкций).

**Что требуется.** Разработке подлежит полнофункциональный прототип (образец) наземной малой мобильной платформы (робот), предназначенной для работы на слабопересеченной местности. Робот должен действовать в составе большой группы подобных устройств в автономном режиме (режим телеуправления является опциональным). Назначение роботов заключается в перемещении в некоторую область и выполнении в ней целевых функций (возврат на базу тоже может быть опциональным). Отсюда следуют такие требования, как максимальная дешевизна таких устройств и простота конструкций. Ограниченность возможностей системы управления робота должна компенсироваться массовостью группировки.

**Условия разработки.** Создание прототипа должно занимать строго определенное время. Полный цикл — от получения технического задания до проведения испытаний прототипа — должен занимать 4 недели. Отсюда ясна необходимость наличия библиотеки типовых решений, а также минимизации времени на разработку и согласование моделей всех уровней проектирования.

**Коллектив.** Работы в каждом случае ведутся коллективом фиксированной структуры. В него входят два программиста-математика, инженер-схемотехник, два инженера-конструктора и руководитель подразделения.

**Этапы разработки.** Они достаточно очевидны. Реализуются математическая, имитационная и физическая модели РТС, разрабатывается прототип робота, проводятся эксперименты, испытывается робот. После получения технического задания инженер-схемотехник и инженеры-конструкторы начинают создавать робота, используя типовые модули. В течение первой недели разрабатывается математическая модель, а также формируются модули для физического моделирования. Вторая неделя — разработка программных интерфейсов и плотная работа с физической моделью. Третья неделя — проведение лабораторных и натурных экспериментов, доработка моделей. Четвертая неделя — создание технической документации и проведение испытаний.

При условии использования типовых решений и стандартных программно-аппаратных модулей в процессе разработки наиболее критичным становится быстрый, желательно с минимальными трудозатратами и потерями, переход между математическими, имитационными и физическими моделями.

Далее в этой работе мы рассмотрим, какие подходы позволят обеспечить максимально простой переход между ними.

## 2. ПРИМЕР ПРОЕКТА

Предположим, что нашей целью является создание группировки наземных мобильных роботов, которые должны доставить некоторую полезную нагрузку в заданный район или область пространства. При этом роботам задаются целевой пеленг и примерное удаление от точки старта. Сами роботы оснащены лишь дальномерами или локаторами, способными определить препятствия или непроходимые участки (ямы), а также средствами локальной связи для определения состояния соседей — роботов своей группы. Ни-

каких систем глобальной навигации и карты местности у них не предполагается. Все, на что могут ориентироваться роботы, — это анализ локального окружения. При этом предположим, что роботы должны двигаться компактной группой, стараясь избегать столкновений и обходя препятствия, выдерживая при этом заданный целевой пеленг.

Последовательность этапов реализации такого проекта очевидна:

- 1) разработка математического обеспечения, моделей, методов,
- 2) разработка демонстрационной имитационной модели для отладки математического аппарата и алгоритмов,
- 3) создание физической модели системы,
- 4) разработка натурной модели — прототипа устройства, то есть робота.

Основная задача заключается в том, чтобы сделать переходы между имитационной, физической и натурной моделями минимально трудозатратными, бесшовными.

### 3. МОДЕЛИ

#### 3.1. Математическая модель

##### 3.1.1. Метод потенциальных полей

Одним из очевидных механизмов реализации как индивидуального, так и группового движения является использование метода потенциальных полей. Фактически речь идет о реализации законов согласованного движения группы агентов, а основой этих законов может быть классическая модель движения К. Рейнольдса [1]. Правила движения агентов заключаются в том, что агент (а) не должен приближаться близко к другим агентам, (б) двигаться туда, куда все, и (в) стремиться не отдаляться от других. Кроме того, к ним добавляется стремление придерживаться заданного направления движения (целевой пеленг), а также не приближаться к препятствиями и/или опасным областям.

Формально это означает, что итоговое направление движения агента определяется как сумма сил отталкивания-притяжения  $\vec{F}_{res}$  вида

$$\vec{F}_{res} = \vec{F}_{goal} + \vec{F}_{sep} + \vec{F}_{align} + \vec{F}_{coh} + \vec{F}_{obst}. \quad (1)$$

Здесь  $\vec{F}_{goal}$  — вектор, определяющий движение по целевому пеленгу,  $\vec{F}_{sep}$  — сила отталкивания от соседей (избегание столкновений),  $\vec{F}_{align}$  — сила выравнивания курса (придерживаться среднего курса соседей),  $\vec{F}_{coh}$  — сила притяжения к своим (ориентация на среднее положение соседей),  $\vec{F}_{obst}$  — сила отталкивания от препятствий.

Закон (1) можно естественным образом расширять путем добавления новых факторов, определяющих тенденцию движения агента.

Все, на что ориентируется агент при реализации правил поведения, — это множество наблюдаемых объектов  $\Omega_s = \{\omega_i^k\}$ , попадающих в некоторую область видимости. Здесь  $\omega_i^k$  — распознаваемые сенсорикой робота объекты — соседи, препятствия, целевые объекты (если таковые есть) и т. п. Объекты задаются как  $\omega_i^k = \langle (x_i, y_i), k \rangle$ , где  $(x_i, y_i)$  — координаты,  $k$  — класс, которому принадлежит объект (соседи, препятствия, целевые объекты и т. п.). Каждый элемент  $\omega_i^k$  определяет свой вклад (знак и величину) итогового вектора движения. В результате итоговая программа движения агента  $Prog(agent)$  определяется значением компонент целевого вектора  $\vec{F}_{res}$ :

$$Prog(agent) = Prog(\vec{F}_{res}) = (v_{ctl}, a_{ctl}), \quad (2)$$

где  $v_{ctl}$  — линейная скорость движения, а  $a_{ctl}$  — угол ориентации.

### 3.1.2. Поведение

Очевидно, что метод потенциальных полей может определить лишь локальный характер движения агента, его текущую реакцию. Исходная задача движения по пеленгу с обходом препятствий — это уже сложная поведенческая программа, для которой требуются иные модели. Одним из удобных и наглядных способов задания простого поведения агента робота являются конечно-автоматные модели, например автоматы с выходом. При этом вход автоматов определяется сенсорикой агента, а выходом является выполнение некоторого действия или последовательности действий. Здесь под действием будем понимать реализацию как прямого управления движением (вперед, влево, вправо), так и программу движения агента, задаваемую выражением (2). Будем придерживаться единой методологии мета-автоматного управления, при которой элементарные действия, их последовательности и комплексы последовательностей представлены в виде иерархий соответствующих автоматов. *Мета-автомат* — это автомат Мили, у которого пометки дуг определяют выполняемые агентом процедуры. Выполнение этих процедур заключается в загрузке соответствующих конечных автоматов (КА). Если действиями КА являются активизации низкоуровневых двигательных функций, то такой автомат имеет ранг 0 —  $M^0$ . Мета-автомат ранга 1  $M^1$  — это КА, действия которого заключаются в управлении автоматами  $M^0$ , и т. д. В итоге мета-автомат ранга  $n$   $M^n$  управляет автоматами ранга  $M^{n-1}$ . Таким образом происходит структуризация иерархии управления в автоматном представлении. Более подробно это изложено в работе [2].

На рисунке 1 приведен мета-автомат, реализующий процедуру движения по пеленгу, задаваемому аргументом  $arg$ .

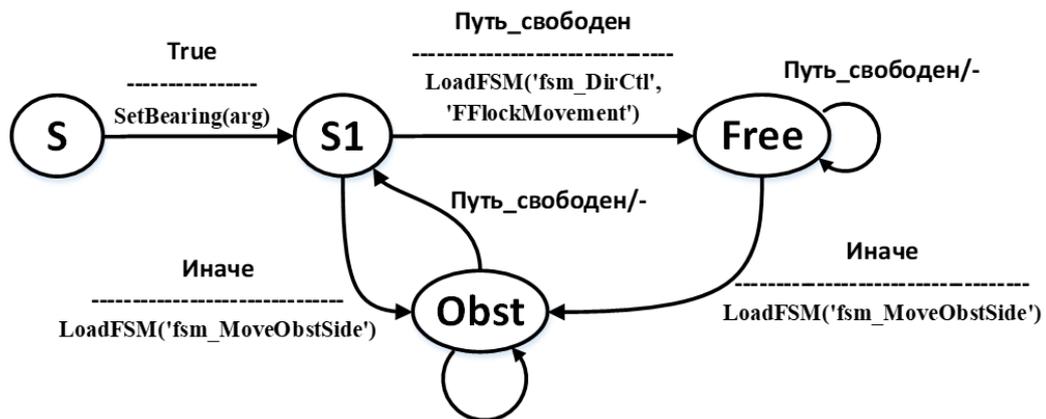


Рис. 1. Мета-автомат уровня 2, реализующий движение по пеленгу

Суть его работы заключается в том, чтобы в зависимости от ситуации активизировать автоматы уровня 1, реализующие либо закон группового движения (автомат  $fsm\_DirCtl$  с аргументом  $FFlockMovement$ ), если путь свободен, либо автомат  $MoveObstSide$ , реализующий движение вдоль обнаруженного препятствия (обход). Автомат  $fsm\_DirCtl$  — это вырожденный автомат с одним состоянием, который реализует двигательную функцию. Автомат  $MoveObstSide$  описывает уже последовательность элементарных движений, определяемых значениями входных сигналов от сенсоров робота.

Особое внимание здесь уделяется отслеживанию наблюдаемых объектов  $\Omega_s$  в области видимости робота, а также компоненту управления движением. Корректное форми-

рование  $\Omega_s$  и адекватность управления  $Prog(agent)$  и определяют в конечном итоге непрерывность перехода от математической модели к имитационной, физической и натурной.

### 3.2. Имитационная модель

Далее строится имитационная модель. Ее задача — показать работоспособность методов и алгоритмов на большом количестве быстрых экспериментов. Для этого используются представления, основанные, например, на геометрических или кинематических моделях для упрощения реализации сенсорики. Такое моделирование является достаточно абстрактным, но в рамках нашей задачи основная цель такого моделирования — предварительная отладка поведенческих процедур. Ниже представлены примеры такого моделирования: с использованием системы Kvorum (рис. 2а), которая была разработана для моделирования поведения больших групп агентов, а также в системе визуализации состояния агентов для отработки моделей движения в трехмерном пространстве (рис. 2б, проект «Рыба»).

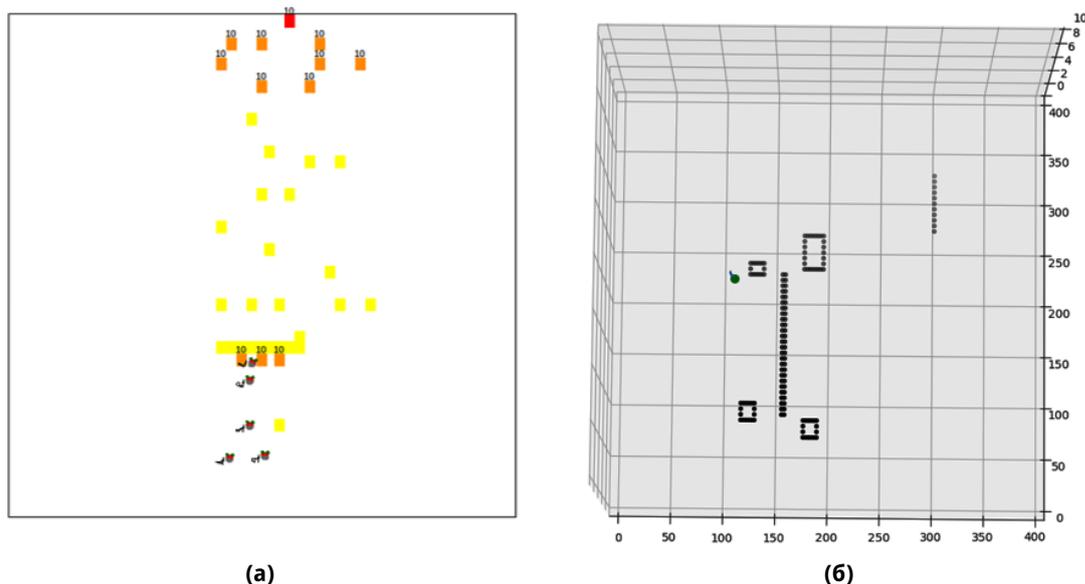


Рис. 2. а) система имитационного моделирования Кворум, б) локальная система моделирования для проекта «Рыба»

### 3.3. Физическая модель

Следующий этап — построение модели, более близкой к реальным условиям. Сейчас существует множество разных систем для имитационного моделирования, например, Microsoft Robotics Developer Studio [3], Webots [4] и др. В данном случае используется одно из распространенных средств для реализации физического моделирования в робототехнике — система Gazebo [5].

Система Gazebo предоставляет мощный инструментарий для создания моделей роботов, поддерживая в том числе геометрические представления из САПР типа Solid Works. Как правило, на начальном этапе прототипирования модель создается именно в САПР, где указываются ее физические характеристики: распределение масс, упругие

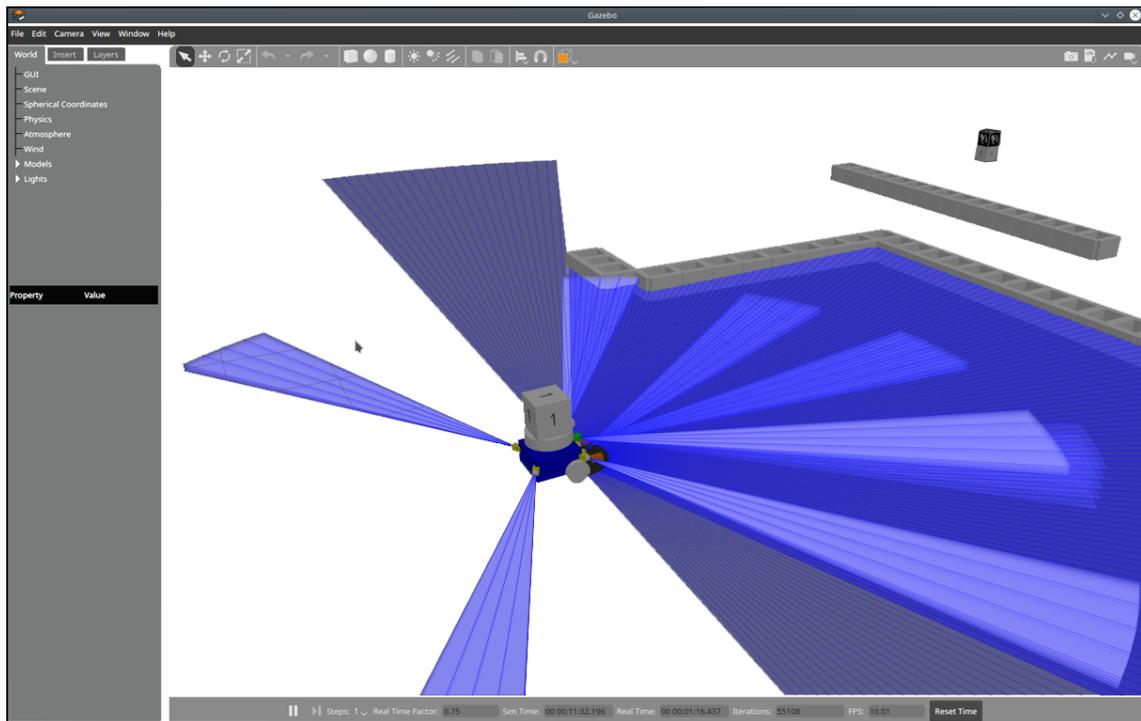


Рис. 3. Пример модели робота в среде Gazebo

свойства, влияние трения и т.п. Далее она экспортируется в файл описания модели робота, который по необходимости дополняет инженер-программист. Это достаточно трудоемкий процесс, требующий определенной квалификации и опыта от инженера. Более того, создание подробных и реалистичных моделей (включая модель внешней среды) зачастую требует времени, превышающего время создания и отладки физического робота. Вместе с тем для задач моделирования поведенческих функций достаточно использования упрощенных моделей роботов. В таком случае не требуется применения САПР: инженеру достаточно самому создать упрощенный файл-описание робота, где корпус представлен, например, в виде параллелепипеда, к которому подсоединены колеса и необходимые датчики. Помимо непосредственного описания блоков, требуется определить места их сочленений, подключить эмуляцию контроллера и сенсорную.

При этом основная сложность заключается именно в описании номенклатуры датчиков робота, так как правильное формирование  $\Omega_s$  является существенным моментом всего процесса. В примере на рисунке 3 показана модель робота с 7 ультразвуковыми дальномерами, серволокатором и камерой, которые формируют  $\Omega_s$ .

Назначение такой модели, помимо предоставления более приближенных к реальности сенсорных данных, — это отладка программных интерфейсов между управляющей программой и реальным роботом. В данном случае проводятся: тестирование корректности представления  $\Omega_s$  на верхнем уровне управления, передача управляющих команд на нижний уровень, сопоставление ожидаемого и получаемого результатов управления.

Условно процесс разработки представлен на рисунке 4.

Интересно отметить, что этап физического моделирования иногда не является обязательным. В идеале, если есть хорошо отлаженная и адекватная поведенческая модель, можно обойтись имитационным моделированием. Ярким примером такого подхода яв-



Рис. 4. Схема процесса разработки

ляются некоторые исследования в области моделей социального поведения роботов. Так, в работе И. П. Карповой [6] описывается реализация модели пространственной ориентации агентов, которая была реализована на имитационном уровне (система Kvorum [7]), а далее эта же модель применялась для управления реальными роботами.

Рассмотрим далее особенности заключительного этапа — создания прототипов роботов.

### 3.4. Создание прототипа

Платформы представляют собой колесные шасси, на которых расположены одинаковые блоки управления и однотипный базовый набор сенсоров. Тестировались два варианта исполнения конструкции — трехколесная платформа с двумя ведущими колесами и полноприводная четырехколесная платформа переломного типа. Платформы обладают принципиально различающимся характером движения, однако, что крайне важно, законы управления для них применяются одинаковые.

Основным компонентом системы управления является RaspberryPi 4. Имеются также ходовой контроллер, контроллеры управления сенсорами и исполнительными устройствами, драйверы и система питания. Сенсорика представлена ультразвуковыми дальномерами (8 УЗД), серволокатором с углом 180°, а также камерой.

#### 3.4.1. Трехколесная платформа

Это классическая система с дифференциальным приводом. На платформе спереди установлены два ведущих колеса, а сзади находится ролевое колесо. Ведущие колеса обеспечены пружинной подвеской. Несмотря на простоту такой схемы, не требующей амортизаторов, небольшой вес конструкции не позволяет ей входить в ощутимые коле-

бания. Переднеприводная схема не обеспечивает хорошей курсовой устойчивости, однако в силу специфики решаемых задач здесь важнее проходимость.

Основная задача платформы — отработка управления в лабораторных условиях, то есть движение на твердых ровных поверхностях с минимальными препятствиями, предельная высота которых определяется диаметром колес. Аккумулятор расположен над передней осью, что обеспечивает нужное распределение весов. Общий вид платформы без целевой нагрузки представлен на рис. 5.

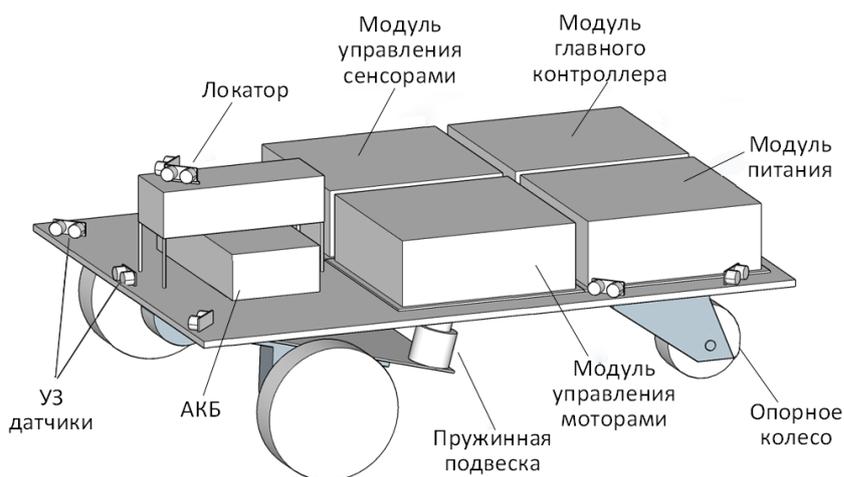


Рис. 5. Общий вид трехколесной платформы. Целевая нагрузка не показана

### 3.4.2. Четырехколесная платформа

Платформа имеет колесную формулу  $4 \times 4$ , при этом все приводы являются независимыми. Платформа обладает проходимостью и устойчивостью, достаточными для движения на слабопересеченной местности. Для обеспечения постоянного сцепления колес с поверхностью и повышения маневренности платформы со схемой  $4 \times 4$  была применена схема с подвижной рамой.

Рама робота состоит из двух полурам, соединенных шарнирным устройством. Полурамы робота могут поворачиваться относительно друг друга вокруг горизонтального и вертикального шарниров. Такая компоновка напоминает техническое решение, описанное в [8] и обеспечивает хорошую маневренность и постоянное зацепление всех колес с почвой. Привод колес осуществляется четырьмя мотор-редукторами, закрепленными на полурамах. Поворот полурам относительно друг друга выполняет линейный актуатор. Общий вид и принцип поворота показан на рисунке 6.

На поворотном узле установлен датчик угла поворота, что обеспечивает обратную связь с актуатором. Модуль управления актуатором никак не затрагивает нижний уровень управления, так как использует свои отдельные программные интерфейсы. Поскольку посередине расположен поворотный узел, система управления скомпонована в два модуля, расположенные спереди и сзади: силовой модуль (питание и управления моторами) и модуль контроллеров.

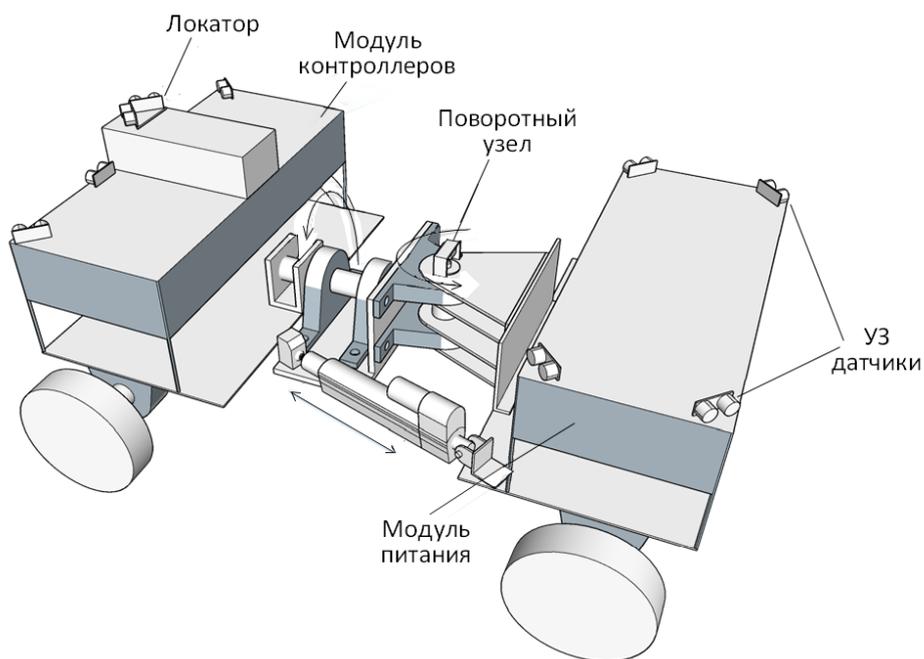


Рис. 6. Общий вид переломной платформы

### 3.4.3. Испытания

Обе платформы, как и условия их эксплуатации, являются принципиально разными, но их различия никак не затрагивают архитектуру и принципы управления ими. Платформы показали достаточную управляемость для решения поставленной задачи. На рисунке 7 представлены описанные выше прототипы изделий без установленной целевой нагрузки.



Рис. 7. Группа малых роботов, в центре — «переломная» платформа

Платформы решали задачу перемещения в заданную область пространства с обходом препятствий, используя только бортовую сенсорную систему без средств глобальной навигации.

#### 4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Особенности формирования личного состава трудовых подразделений и характера работ требуют применения специфических форм их обучения, подачи материала. Главная задача обучения — сформировать у личного состава навыки использования нужных им инструментов для быстрого получения работоспособного продукта. Исходя из практической направленности курса, наилучшей моделью его организации является проектный подход [9], позволяющий наглядно продемонстрировать основные этапы разработки и используемые в процессе инструменты [10]. Здесь в полной мере должны использоваться присущие робототехнике наглядность, предметность и кажущаяся простота решаемых задач, за которыми обычно скрыты значительные сложности в их формализации. Большой объём теоретического материала для данного формата не требуется, так как предполагается наличие у обучаемых предварительной подготовки — умения разрабатывать алгоритмы и реализовывать их программно. В качестве основного языка программирования был использован распространённый и легко осваиваемый Python, что расширяет круг потенциальных обучаемых. Хотя сам по себе язык Python и не лучший вариант для обучения программированию высоконадёжных и высоконагруженных систем [11], но такое решение позволяет значительно ускорить разработку прототипа. В дальнейшем часть ПО может быть переписана на C++.

Курс состоит из десяти практических работ. Восемь из них — это работы базового уровня, в каждой из которых требуется дополнить и изменить одну и ту же модель мобильного робота, что позволяет постепенно расширять его возможности, каждый раз демонстрируя их учащимся на имитационной модели. Две работы — это специализированные занятия, посвященные особенностям использования целевой нагрузки. После завершения практических работ курсанты под руководством инструктора разрабатывают свои варианты решения поставленной «транспортной задачи».

Разумеется, этот подход в данной области нельзя назвать принципиально новым, потому что существует множество учебных курсов разного уровня, решающих те же задачи, что и предлагаемый, также через проектный подход. По характеру усваиваемых навыков близкими аналогами можно назвать вводные занятия по введению в робототехнику наподобие [12, 13]. Основное отличие данной системы обучения состоит в последовательном и специфическом применении моделей разного уровня, формирующих у курсантов необходимый для робототехника навык структурного и поведенческого моделирования системы. Другой важной особенностью является лёгкость варьирования аппаратной части благодаря использованию унифицированных и легко поддерживаемых ROS-интерфейсов. Официально рекомендуемые обучающие материалы ROS и вводные книги [14] могут дать обучаемым некоторые знания в данной области, но не обеспечивают практического применения полученных результатов либо переноса разработок между моделями разного уровня.

Апробация учебных курсов проводилась среди студентов бакалавриата младших и старших курсов ряда вузов, обучающихся на разных, не только инженерно-технических, направлениях. Именно контингент подобного рода и составляет целевые рабочие подразделения. Характерным оказалось то, что наличие робототехнической специализации обучения практически никак не влияло на качество усвоения материала и получения практических навыков. Это — следствие специфики курсов, направленных, прежде всего, на решение задач управления.

## 5. ЗАКЛЮЧЕНИЕ

Стремление к созданию «беспшовных» переходов между различными видами моделей является вполне естественным и очевидным. Для этого и создают различного рода комплексы и пакеты моделирования, которые позволяют согласовывать интерфейсы, связывать между собой компоненты, следить за их совместимостью, как Nvidia Isaac [15, 16] и др. Основное внимание при этом уделяется вопросам удобного взаимодействия с аппаратурой, а более высокоуровневые задачи предлагается решать иными средствами. В предлагаемой системе внимание сфокусировано на несколько ином аспекте — на том, что может дать подход, основанный на поведенческом моделировании.

Итак, выделим основные особенности реализации быстрого проектирования специализированных мобильных робототехнических систем:

1. Минимизация трудозатрат при переходе от моделей разного уровня (обеспечение «беспшовности») может быть достигнута путем использования в системах управления робота высокоуровневых и абстрактных поведенческих моделей. Эти модели имеют минимальную привязку к реальным абсолютным физическим величинам, а оперируют выполнениями различных действий в зависимости от текущих условий.
2. При таком подходе важно иметь библиотеки решений для различных этапов моделирования: на уровне имитационного моделирования — библиотеку метавтоматов, на уровне физического моделирования — унифицированных модулей и драйверов, на конструкторском уровне — соответствующие физические и конструкционные узлы.

Иными словами, быстрая реализация, поставленная на поток, требует применения своего рода конструктора, способного реализовывать различные поведенческие модели на более высоком уровне, чем это позволяет делать базовый вариант ROS. Все прочие технологии и этапы проектирования, естественно, остаются неизменными. Очевидно, что решения, построенные из унифицированных, типовых блоков, всегда будут проигрывать в качестве специализированным решениям, однако повторим, основными критериями качества для решаемой задачи являются время и массовость такого производства. Тем более что изначально упор делался на разработку не единичных роботов, а их группировок — простых, дешевых и многочисленных.

### Благодарности

Работа проведена в рамках выполнения государственного задания НИЦ «Курчатовский институт».

### Список литературы

1. Reynolds C. W. Flocks, herds, and schools: A distributed behavioral model // Proc. 14th Annu. Conf. Comput. Graph. Interact. Tech. SIGGRAPH 1987. 1987. Т. 21. № 4. С. 25–34.
2. Карпов В. Э., Повбо М. А., Воробьев В. В. О некоторых аспектах применения автоматных моделей в групповом управлении // Мехатроника. Автоматизация. Управление. 2023. Т. 24, № 4. С. 171–180. doi:10.17587/mau.24.171-180
3. Microsoft Robotics Developer Studio 2019 [Электронный ресурс]. URL: <https://learn.microsoft.com/en-us/archive/msdn-magazine/2008/june/robotics-simulating-the-world-with-microsoft-robotics-studio> (дата обращения: 18.08.2024).
4. Cyberbotics Ltd. Webots. 2023 [Электронный ресурс]. URL: <http://www.cyberbotics.com/> (дата обращения: 18.08.2024).

5. Gazebo, Official Resource. 2024 [Электронный ресурс]. URL: <https://gazebosim.org/> (дата обращения: 11.08.2024).
6. Карпова И. П. Об одном биоинспирированном подходе к ориентации роботов, или настоящий «муравьиный» алгоритм // Управление большими системами. 2022. Т. 96. С. 69–117. doi:10.25728/ubs.2022.96.5
7. Карпов В. Э., Ровбо М. А., Овсянникова Е. Е. Система моделирования поведения групп робототехнических агентов с элементами социальной организации Кворум // Программные продукты и системы. 2018. Т. 31, № 3. С. 581–590. doi:10.34220/issn.2222-7962/2020.4/18
8. Клубничкин В. Е., Клубничкин Е. Е., Горбунов А. Ю., Дручинин Д. Ю. Разработка узла сочленения лесной погрузочно-транспортной машины // Лесотехнический журнал. 2021. Т. 10, № 4. С. 217–226. doi:10.1177/016146811801900404
9. Kilpatrick W. H. The Project Method // Teachers College Record: The Voice of Scholarship in Education 1918. Т. 19, № 4. С. 1–5. doi:10.1177/016146811801900404
10. Уракова Е. А., Быстрова Н. В., Грашина П. А. Сущность проектного подхода в профессиональном образовании // Проблемы современного педагогического образования. 2020. Т. 69, № 4. С. 276–278.
11. Павлов Д. А. Язык Python в высшем образовании // Компьютерные инструменты в образовании. 2024. № 1. С. 85–94. doi:10.32603/2071-2340-2024-1-85-95
12. Asada H., Leonard J. Introduction To Roboticsю 2005 [Электронный ресурс]. URL: <https://ocw.mit.edu/courses/2-12-introduction-to-robotics-fall-2005/> (дата обращения: 18.08.2024).
13. Asokan T., Ravindran B., Vasudevan K. Introduction to roboticsю 2024 [Электронный ресурс]. URL: [https://onlinecourses.nptel.ac.in/noc20\\_de11/preview](https://onlinecourses.nptel.ac.in/noc20_de11/preview) (дата обращения: 18.08.2024).
14. Lentin J., Johnny A. Robot Operating System for Absolute Beginners. New York: Apress, 2022. 295 с.
15. NVidia. NVidia Isaакю [Электронный ресурс]. URL: <https://developer.nvidia.com/isaac> (дата обращения: 18.08.2024).
16. RoboDK Моделирование и программирование в RoboDK. 2024 [Электронный ресурс]. URL: <https://robodk.com/ru/simulation> (дата обращения: 18.08.2024).

Поступила в редакцию 19.08.2024, окончательный вариант — 27.09.2024.

**Карпов Валерий Эдуардович, доктор технических наук, доцент, начальник лаборатории робототехники, НИЦ Курчатowski институт, [Karpov\\_VE@nrcki.ru](mailto:Karpov_VE@nrcki.ru)**

**Сорокоумов Петр Сергеевич, научный сотрудник, НИЦ Курчатowski институт, [Sorokoumov\\_PS@nrcki.ru](mailto:Sorokoumov_PS@nrcki.ru)**

**Карпов Валерий Валерьевич, инженер-исследователь, НИЦ Курчатowski институт, [Karpov\\_VV@nrcki.ru](mailto:Karpov_VV@nrcki.ru)**

**Воробьев Виталий Владимирович, заместитель начальника лаборатории робототехники, НИЦ Курчатowski институт, ✉ [Vorobev\\_VV@nrcki.ru](mailto:Vorobev_VV@nrcki.ru)**

Computer tools in education, 2024

№ 3: 79–92

<http://cte.eltech.ru>

doi:10.32603/2071-2340-2024-3-79-92

## On One Approach to Organizing Rapid Prototyping of Robots under Rigid Constraints

Karpov V. E.<sup>1</sup>, Doctor sc., Docent, [Karpov\\_VE@nrcki.ru](mailto:Karpov_VE@nrcki.ru)

Sorokoumov P. S.<sup>1</sup>, researcher, [Sorokoumov\\_PS@nrcki.ru](mailto:Sorokoumov_PS@nrcki.ru)

Karpov V. V.<sup>1</sup>, research-engineer, [Karpov\\_VV@nrcki.ru](mailto:Karpov_VV@nrcki.ru)

Vorobiev V. V.<sup>1</sup>, deputy head of laboratory, ✉ [Vorobev\\_VV@nrcki.ru](mailto:Vorobev_VV@nrcki.ru)

<sup>1</sup>NRC Kurchatov Institute, 1 Kurchatov Square, 123182, Moscow, Russia

### Abstract

The paper describes an approach to the rapid prototyping of ground-based group robotic systems by small teams of developers in a short time. Prototyping is carried out by sequentially constructing representations of various levels of abstraction. It is demonstrated that the main resource costs in the transition between these levels can be reduced by utilizing standard behavioral models of robots and applying libraries of typical software design solutions. Behavioral models based on the mechanism of meta-automata can use abstract implementations at the mathematical, simulation and physical levels to obtain quite practical and efficient solutions. ROS is used as the basic tool for developing control systems and conducting physical modeling on lower levels. The developed approach is demonstrated on the example of a transport problem solved by robots with the simplest sensors without global navigation.

**Keywords:** *group robotics, modeling, behavior models, project-based learning.*

**Citation:** V. E. Karpov, P. S. Sorokoumov, V. V. Karpov, and V. V. Vorobiev, "On One Approach to Organizing Rapid Prototyping of Robots under Rigid Constraints," *Computer tools in education*, no. 3, pp. 79–92, 2024 (in Russian); doi:10.32603/2071-2340-2024-3-79-92

### References

1. C. W. Reynolds, "Flocks, herds, and schools: : A distributed behavioral model," in *Proc. 14th Annu. Conf. Comput. Graph. Interact. Tech. SIGGRAPH 1987*, vol. 21, no. 4, pp. 25–34, 1987.
2. V. E. Karpov, V. V. Vorobiev, and M. A. Rovbo, "About Some Aspects of Finite State Machine Models Application to Group Control," *Mekhatronika, Avtomatizatsiya, Upravlenie*, vol. 24, no. 4, pp. 171–180, 2023 (in Russian); doi:10.17587/mau.24.171-180
3. Microsoft Corp., "Microsoft Robotics Developer Studio," in *learn.microsoft.com*, 2019. [Online]. Available: <https://learn.microsoft.com/en-us/archive/msdn-magazine/2008/june/robotics-simulating-the-world-with-microsoft-robotics-studio>
4. Cyberbotics Ltd., "Webots Open Source Robot Simulator," in *www.cyberbotics.com*, Jun. 2023. [Online]. Available: <http://www.cyberbotics.com/>
5. Gazebo, "Official Resource," in *gazebosim.org*, 2024. [Online]. Available: <https://gazebosim.org/>
6. I. P. Karpova, "A bioinspired approach to robot orientation or a real «ant» algorithm," *UBS*, vol. 96, pp. 69–117, 2022 (in Russian); doi:10.25728/ubs.2022.96.5

7. V. E. Karpov, M. A. Rovbo, E. E. Ovsyannikova, “Kvorum – the system for modeling the behavior of robotic agent groups with elements of social organization,” *Research Institute Centerprogramsystem*, vol. 31, no. 3, pp. 581–590, 2018 (in Russian).
8. V. Klubnichkin, E. Klubnichkin, A. Gorbunov, and D. Druchinin, “Development of the forwarder articulation joint,” *Forestry Engineering Journal*, vol. 10, no. 4, pp. 217–226, 2021 (in Russian); doi:10.34220/issn.2222-7962/2020.4/18
9. W. H. Kilpatrick, “The Project Method,” *Teachers College Record: The Voice of Scholarship in Education*, vol. 19, no. 4, pp. 1–5, 1918, doi:10.1177/016146811801900404
10. E. A. Urakova, N. V. Bystrova, and P. A. Grashina, “Essence of design approach in vocational education,” *Problems of modern pedagogical education*, vol. 69, no. 4, pp. 276–278, 2020 (in Russian).
11. D. Pavlov, “Should we teach Python at universities?,” *Computer tools in education*, no. 1, pp. 85–94, 2024 (in Russian); doi:10.32603/2071-2340-2024-1-85-95
12. H. Asada and J. Leonard, “Introduction To Robotics,” in *ocw.mit.edu*, 2005. [Online]. Available: <https://ocw.mit.edu/courses/2-12-introduction-to-robotics-fall-2005/>
13. T. Asokan, B. Ravindran, and K. Vasudevan, “Introduction to robotics,” in *onlinecourses.nptel.ac.in*, 2024. [Online]. Available: [https://onlinecourses.nptel.ac.in/noc20\\_de11/previe](https://onlinecourses.nptel.ac.in/noc20_de11/previe)
14. J. Lentin and A. Johnny *Robot Operating System for Absolute Beginners*, New York: Apress, 2022.
15. NVidia Corp., “NVidia Isaac,” in *developer.nvidia.com*, 2024. [Online]. Available: <https://developer.nvidia.com/isaac>
16. RoboDK Global, “Simulate and Program your Robot in 5 easy steps,” in *robodk.com/ru/simulation*, 2024. [Online]. Available: <https://robodk.com/ru/simulation>

*Received 19-08-2024, the final version — 27-09-2024.*

**Valery E. Karpov, Doctor of Sciences (Tech.), Docent, head of robotics laboratory, NRC Kurchatov Institute, [Karpov\\_VE@nrcki.ru](mailto:Karpov_VE@nrcki.ru)**

**Petr Sorokoumov, researcher, NRC Kurchatov Institute, [Sorokoumov\\_PS@nrcki.ru](mailto:Sorokoumov_PS@nrcki.ru)**

**Valery V. Karpov, research-engineer, NRC Kurchatov Institute, [Karpov\\_VV@nrcki.ru](mailto:Karpov_VV@nrcki.ru)**

**Vitaly Vorobiev, deputy head of laboratory of robotics, NRC Kurchatov Institute, [✉ Vorobev\\_VV@nrcki.ru](mailto:Vorobev_VV@nrcki.ru)**