

УДК 519.878, 519.1, 004.942

Евгений Аркадьевич Магид (magid@it.kfu.ru), Роман Олегович Лавренов (lavrenov@it.kfu.ru), Артур Газизович Сагитов (sagitov@it.kfu.ru)

(Казанский (Приволжский) Федеральный Университет, Россия, г. Казань)

Максим Алексеевич Соколов (m.sokolov@innopolis.ru), Илья Михайлович Афанасьев (i.afanasyev@innopolis.ru)

(Университет Иннополис, Россия, г. Иннополис)

ОПЫТ РАЗРАБОТКИ СИМУЛЯТОРА ДЛЯ МОБИЛЬНОГО РОБОТА «СЕРВОСИЛА «ИНЖЕНЕР»*

Цель большей части исследований в робототехнике – заменить людей в тех случаях, когда выполняемые ими действия слишком примитивны или однообразны, или же наоборот, сложны и опасны. Например, при проведении поисково-спасательных работ в ситуации чрезвычайного происшествия или техногенной аварии, а также в случаях необходимости производства разведки и исследований в условиях, когда окружающая обстановка или агрессивные условия окружающей среды не позволяют человеку выполнить поставленную задачу. Для подобных задач предпочтительнее использовать гусеничные роботы, как более проходимые в сравнении с колесными. В данной статье представлены наши шаги по моделированию российского гусеничного робота «Инженер» компании «Сервосила» в среде ROS/Gazebo. Движение робота и взаимодействие его видимых частей визуализируется в программном обеспечении RViz. При создании симулятора использовались оригинальные САД-модели частей робота от производителя. В этой статье мы представляем подходы для моделирования движения гусениц и представляем собственный метод симуляции робота, включая его манипулятор и захват, а также работу бортовых датчиков. Статья рассматривает современный мировой опыт подобных разработок, способы, которые использовались нами при создании модели, и проблемы, с которыми мы столкнулись. В заключительной части статьи представлено тестирование модели в рамках гетерогенной группы роботов в сценарии одновременной локализации, картографирования и планирования пути в среде симуляции Gazebo на неизвестной карте. Алгоритмы расчета пути для гетерогенной группы беспилотных наземных роботов (БНР) и беспилотных летательных аппаратов (БПЛА), разработанные нами ранее, включают в себя комбинацию глобальных и локальных методов построения маршрута.

Поисково-спасательные операции, ROS/Gazebo, моделирование роботов, симуляция гусениц, системная интеграция, мобильный робот.

Evgeni Magid (magid@it.kfu.ru), Roman Lavrenov (lavrenov@it.kfu.ru), Artur Sagitov (sagitov@it.kfu.ru)

(Kazan (Volga region) Federal University, Russian Federation, Kazan)

Maxim Sokolov (m.sokolov@innopolis.ru), Ilya Afanasyev (i.afanasyev@innopolis.ru)

(Innopolis University, Russian Federation, Innopolis)

MOBILE ROBOT "SERVOSILA "ENGINEER" SIMULATOR DEVELOPMENT

The goal of most research in robotics is to replace people in those cases where required actions are primitive and/or monotonous, or vice versa, are too complicated and/or too dangerous. For example, in urban search and rescue (USAR) operations, in case of emergencies or operations in inaccessible to a human locations. In such situation, it is preferable to use a crawler robot due to its higher maneuverability and terrain crossing capacity as compared to a typical wheeled robot. This paper introduces our approach of modeling Russian crawler robot "Engineer" of "Servosila" company in ROS/Gazebo environment. We describe existing approaches to crawler robot modelling and problems that we encountered during modelling process. Robot locomotion and interaction of its visible parts is rendered and visualized in RViz software. Original CAD-models of robot "Engineer"

* Работа выполнена при финансовой поддержке РФФИ (проект № 15-57-06010).

were obtained from the manufacturer and were used for building the simulator. In this paper, we present solutions for modeling caterpillar locomotion and present our simulation of the robot, which covers its base part with main crawlers and additional flippers, a manipulator, a gripper and onboard sensors. Finally, we test the proposed model in heterogeneous robotic group navigation scenario within uncertain Gazebo environment for an unknown or partially known map. Algorithms for calculating a path for heterogeneous group of an unmanned ground vehicle (UGV) and a pair of unmanned aerial vehicles (UAV) were developed earlier by our team and include a combination of global and local methods of route construction.

USAR, ROS/Gazebo, robot modelling, track simulation, system integration, mobile robot.

Введение. При проведении разведывательных и поисково-спасательных операций, особенно в ситуациях, когда в процессе выполнения миссии существует высокий риск травмы или смерти человека, использование беспилотного наземного робота предпочтительнее, чем применение стандартной команды, состоящей из людей и собак. Примерами таких сценариев являются поиск пострадавших в зданиях с высокой вероятностью обрушения или в задымленных, радиационно-загрязненных объектах, при разминировании городской инфраструктуры или проведении антитеррористических операций. При этом, в целях обеспечения большей проходимости, оптимальным решением становится принятие на вооружение роботов на гусеничной базе, а не на колесной [1].

Несмотря на интерес к мобильной робототехнике гусеничного типа, обзор литературы продемонстрировал отсутствие существенного прогресса по моделированию и визуализации траков гусениц. Отдельные исследования в рамках изучения роботов на гусеничной базе касаются симуляции гусеничного хода [1] или описывают математическое моделирование гусениц [2] со взаимодействием с поверхностями различного типа [3][4].

В нашей работе в качестве автономной платформы используется российский гусеничный робот «Сервосила «Инженер». Оригинальное программное обеспечение робота не содержит функционала автономной работы, а пользователи получают доступ к клиент-серверной системе для управления роботом в режиме телеоперации. Этот режим предусматривает управление роботом на расстоянии, когда оператор на основе изображений с камер робота отправляет команды управления движением; в идеале, управление происходит в режиме реального времени, но фактически скорость и качество управления зависят от возможностей канала связи и условий обмена данными.

Наша основная цель - создать программное обеспечение для частичной автоматизации управления роботом, основанное на фреймворке ROS. Первой важной задачей на пути автоматизации управления является 3D-моделирование робота в симуляторе Gazebo. Интеграция ROS и Gazebo позволяет моделировать робота «Инженер» в трехмерной окружающей среде, внедрять алгоритмы навигации и обеспечивать удобную синхронизацию между 3D-моделью и реальным роботом. Бортовые датчики реального робота также будут эмулироваться в полном объеме их возможностей. Для проверки работоспособности 3D-модели робота, проводится экспортирование в стек навигации, который был разработан нами ранее [5] для навигации от точки к точке в симуляторах ROS/Gazebo/RViz.

В настоящее время одной из самых актуальных проблем поисково-спасательной робототехники является взаимодействие беспилотных наземных роботов (БНР) и беспилотных летательных аппаратов (БПЛА). Гетерогенная группа роботов должна работать как единая команда с целью улучшения получаемых об окружающей среде данных. За счет различных сенсорных систем роботов и различного операционного пространства, правильное сочетание особенностей неоднородных систем в составе группы предоставляет возможность более успешно решать транспортно-логистические, разведывательные и поисково-спасательные задачи. В нашем исследовании мы используем созданную модель робота «Инженер» в составе гетерогенной группы роботов, которая должна проводить локализацию и картографирование окружающего пространства, а также решать задачу планирования маршрута от точки к точке. Мы разрабатываем прототипы алгоритмов навигации для гетерогенной группы роботов в среде

MATLAB; разрабатываемые алгоритмы призваны объединить подходы локального и глобального планирования. Далее алгоритмы моделируются в среде ROS/Gazebo для интенсивного тестирования, что сокращает объем экспериментальных работ и повышает их безопасность [6] [7].

1. Наземный гусеничный робот «Инженер». Мобильный робот «Инженер» - это робот на гусеничном ходу (рис. 1), который разработан российской компанией «Сервосила» специально для поисково-спасательных операций в случае природных и техногенных катастроф, а также для проведения технических и саперных работ, в том числе для поиска мин под днищами военной и гражданской техники, осмотра салонов транспортных средств и других задач.



Робот является водо- и пыленепроницаемым, способен работать в условиях тяжелых условий дождя и снега. Его специальное шасси позволяет преодолевать различные препятствия, которые могут появиться в городской среде после стихийного бедствия. Робот имеет прочный металлический корпус, защищающий внутреннюю электронику и бортовые датчики.

Рис. 1. Робот «Инженер» компании «Сервосила» (Россия)

Общий вес робота без бортовых датчиков составляет около 16,3 кг. Робот способен подниматься по лестнице, проходить через дверные проемы и узкие проходы, проникать в здания и, при необходимости, поднимать при помощи мощного двухзвенного манипулятора «голову», в которой размещено большинство датчиков. Это позволяет оператору осуществлять визуальный осмотр местности и объектов, заглядывая внутрь окон припаркованных транспортных средств, наземных этажей и т.д. Манипулятор с захватом может в режиме дистанционного управления поднимать, толкать и тянуть потенциально опасные объекты или открывать двери. Разнообразие датчиков и сенсоров «Инженера» включает в себя приемник GPS/Глонасс, 3D и/или 2D лазерный сканер, система стереозрения, камера заднего вида, камера переднего обзора с оптическим зумом, блок инерциальных датчиков. Также роботу оснащен мощным светодиодным фонарем. В связи с большим числом степеней свободы, множества сенсоров и довольно мощного бортового компьютера, а также возможностью подбирать комплектацию робота под конкретную задачу пользователя, робот «Инженер» стал популярной научно-образовательной платформой для лабораторий и университетов.

2. Среда разработки и моделирования ROS/Gazebo. Симуляторы роботов позволяют проектировать, моделировать и тестировать робототехнические приложения в необходимой физической среде, независимо от наличия реальных аппаратных средств, что существенно со-

кращает время и уменьшает стоимость разработки. Моделирование робота «Инженер» выполнялось нами в симуляторе Gazebo 2.2.3, который интегрирован с операционной системой ROS версии Indigo. Операционная система для роботов ROS (англ. «Robot Operating System») – это фреймворк с открытым исходным кодом, разработка которого началась в 2007 году. Постепенно ОС ROS превратилась в общую мировую тенденцию для симуляции как отдельных датчиков и роботов, так и для групп роботов и сложных взаимодействий при распределённой работе систем.

ROS был выбран нами для разработки алгоритмов и программного обеспечения робота, так как он поддерживает такие системные сервисы как низкоуровневое управление, аппаратную абстракцию и управление пакетами. Несколько взаимосвязанных процессов, запущенных в ROS могут быть представлены в виде графика, основным элементом которого являются «ноды» (узлы, англ. «node»). Ноды могут принимать и передавать данные от датчиков, актуаторов, и обрабатывать различные сообщения.

Совместно с ROS широко используется симулятор Gazebo, который позволяет создать окружающую среду для мобильного робота, визуализировать работу робота и его взаимодействие с окружающей средой, а также работу бортовых датчиков и камер робота в динамике. Gazebo в качестве физического движка использует Открытый физический движок ODE (англ. «Open Dynamics Engine») и Объектно-ориентированный графический движок OGRE (англ. «Object-Oriented Graphics Rendering Engine») для 3D рендеринга, имеет бесплатную лицензию для научных исследований, предоставляет пользователю удобный графический интерфейс и позволяет описывать динамику робота на языке Питон.

3. Моделирование базовой части робота. 3D-модель робота состоит из нескольких частей:

- 1) Основная часть с базовыми гусеницами и батареей
- 2) Два передних флиппера (т.е. дополнительные передние гусеницы)
- 3) Зажимное соединение для связи верхней части с базой робота
- 4) Двухзвенный трехосевой манипулятор с захватом и основной блок системы сбора данных и управления («голова робота»).

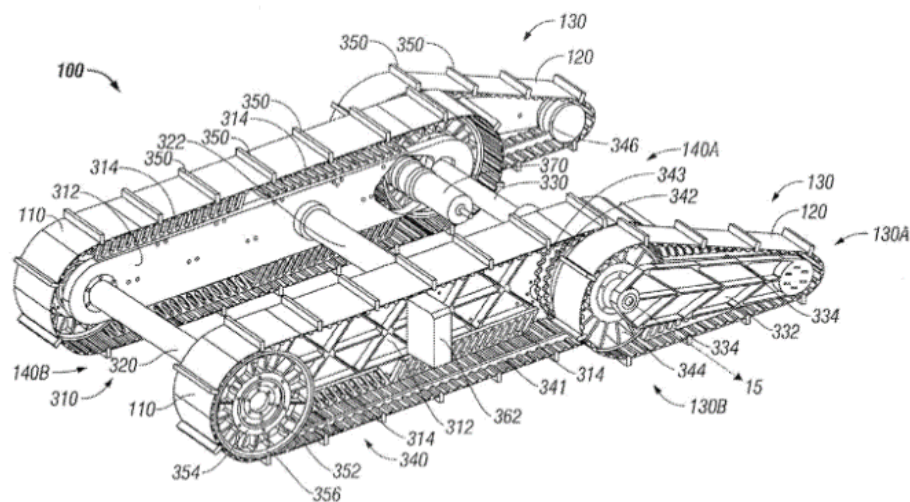


Рис. 2. Схематическое изображение гусеничной базы робота «Инженер»

Для моделирования робота «Инженер» в Gazebo мы использовали полученные от компании «Сервосила» САD-модели и последовательно создали:

- (1) «*.step.» файлы применяя физические параметры частей робота для обмена данными между программными САD-комплексами.
- (2) «*.dae.» файлы для ROS приложений, в которых все части робота были трансформированы в отдельные файлы с соответствующими спецификациям параметрами.
- (3) далее, использовав части модели («*.dae.» файлы), была собрана анимационная модель в формате «*.urdf.» (англ. «Unified Robot Description Format», URDF), в которой все части робота были соединены с помощью специальных ROS-суставов (англ. «ROS joints») в системе

ROS. Для каждого ROS-сустава мы установили ограничения по возможным значениям углов в соответствии со спецификациями робота. Результат моделирования показан на рисунке 3.

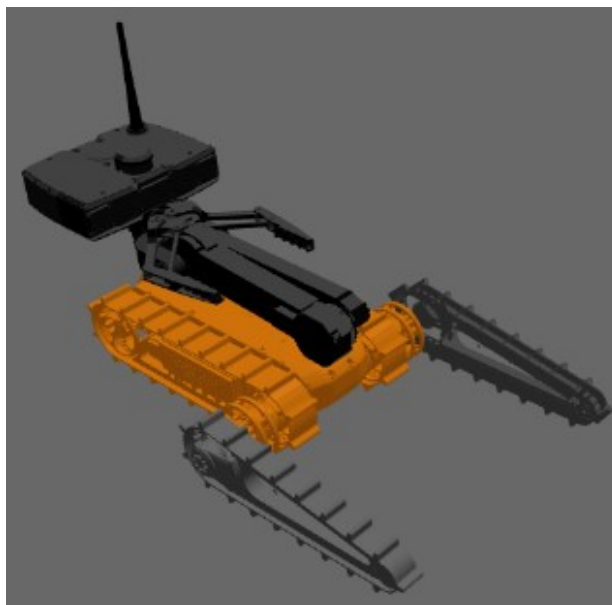


Рис. 3. Визуализация модели URDF робота в симуляторе Gazebo

3.1. Моделирование флипперов и гусениц робота методом псевдоколес. Моделирование механизма ременной передачи для цепочки траков гусениц является сложной задачей, поскольку множество деталей должны работать синхронно. При создании упрощенной модели без физики взаимодействия траков и поверхности, симулирующей постоянный контакт гусениц модели с поверхностью окружающей среды, мы добавили восемь мнимых псевдоколес, по четыре с каждой стороны. Эти колеса располагаются вдоль гусениц робота: одна пара расположена на концах передних флипперов, одна пара на сочленении флипперов с базой робота, одна пара – в центре и еще одна на задней оси крутящего вала робота, при этом две последние пары псевдоколес расположены на расстоянии, равном ширине флиппера, от основных гусениц (рисунок 4).

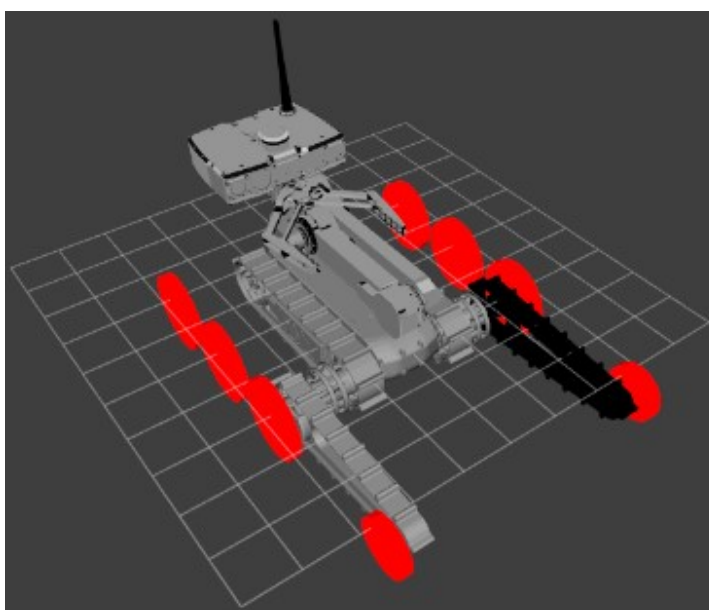


Рис. 4. Робот с псевдоколесами (красные) в RViz

Псевдоколеса соединены с корпусом робота с помощью ROS-суставов с использованием интерфейса управления вращением по углу и по скорости (ROS пакеты `joint_position_controller`, `joint_velocity_controller`). Подобное решение для имитации движения гусениц используют научные группы Института Интеллектуального Анализа и Информационных Систем IAIS Фраунгофера [8] и Технического Университета Дармштадта [1][9]. Увеличение числа мнимых колес вместе с уменьшением их размера увеличивает соответствие модели физике реальной гусеницы. Скорость колес синхронизируется с помощью запросов и ответов через среду ROS. Моделирование движения флипперов использует ROS-суставы для прикрепления флипперов к базе робота, что решило проблему разного размера направляющих роликов на валах, ответственных за движение гусениц, и их согласования с параметрами гусениц. Синхронизация колес выполнена с помощью расчета скорости вращения из соотношения радиусов колес (рисунок 4). Модель способна выполнять подъем и опускание флипперов, движение отдельно правыми и левыми гусеницами вперед и назад.

3.2. Моделирование гусениц с помощью цепочки траков. Решение задачи моделирования гусениц с помощью псевдоколес является упрощенной версией для цепочки гусеничных траков. Трудности в моделировании траков возникают из-за неоднородной структуры ленты и поддерживающих ее элементов, в связи с значительными изменениями свойств при их соединении в цепочку.

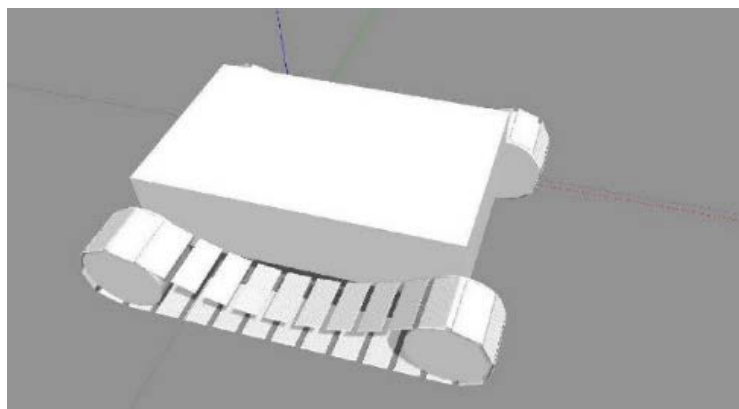


Рис. 5. Прототипирование гусеничного робота, пакет *Gazebo-tracks*

Апробация реалистичной имитации гусеничной ленты в ROS\Gazebo с учетом физики контакта ленты с поверхностью проведена при помощи единственного существующего на сегодня ROS пакета для моделирования подобной ленты – *Gazebo-tracks*. В нем содержится скрипт, создающий гусеничную ленту в виде SDF модели для среды Gazebo [10]. SDF модели используют XML формат для описания объектов и окружающей среды для ROS\Gazebo. Пакет *Gazebo-tracks* поддерживает моделирование гусеницы с двумя равными радиусами роликов и, таким образом, не может быть непосредственно применен для флипперов «Инженера» с направляющими роликами различных радиусов. *Gazebo-tracks* учитывает расстояние между траками, радиус ролика, расстояние между роликами, ширину гусеничной цепи и толщину, число траков в гусенице. Пример визуализации SDF файла гусеничного прототипа робота (основной корпус) показан на рисунке 5.

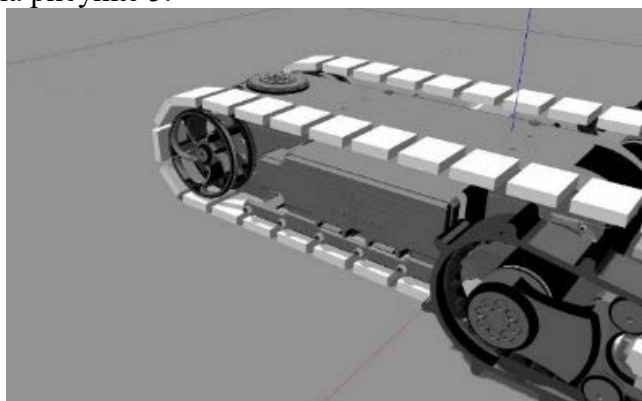


Рис. 6. Результат моделирования гусениц робота «Инженер»

Первоначально нами была создана полная модель робота «Инженер» в формате URDF. Интеграция модели показала, что URDF формат не поддерживает соединений траков в замкнутую цепочку, и только формат SDF может быть применен для моделирования такой цепочки. Использование данного формата позволяет подключать гусеницы из траков к базе робота (рисунок 6), однако, такое решение продемонстрировало ряд критических недостатков:

1) Гусеницы нестабильны и спадают с направляющих роликов вскоре после начала движения в результате воздействия инерционных составляющих и накапливаемого суммарного сдвига траков относительно роликов;

2) Пакет *Gazebo-tracks* поддерживает только мобильные гусеничные системы с одинаковыми радиусами направляющих роликов гусениц;

3) Пакет *Gazebo-tracks* не имеет полноценного ROS API, и, таким образом, ролики не синхронизируются с помощью ROS-топиков и обмена сигналов.

Фактически, выявленные недостатки существующих решений требуют разработки нового подхода моделирования на основе ROS, который будет способен поддерживать цепочки траков любой конфигурации для различного количества, расположения и радиусов опорных катков.

4. Моделирование верхней части робота. Верхняя часть робота состоит из трехосевого манипулятора, захвата и головного блока над ним, и их моделирование проходило по аналогии с созданием модели базы. Смежные части модели связаны между собой посредством ROS-суставов и ограничены по углам в соответствии со спецификациями робота (рисунок 7). При моделировании верхней части возникла проблема управления манипулятором, вызванная распределением веса: захват и головной блок робота весят вместе 2 кг, при общем весе верхней части робота в 4 кг. Таким образом, когда робот поднимает головной блок на максимальную высоту через поворот всех осей манипулятора, возникает существенный крутящий момент из-за большой массы головного блока и длинного рычага манипулятора. Однако, если поворачивать только второй (средний) ROS-сустав, крутящий момент значительно снижается. Для оптимизации управления роботом-манипулятором с ПИД-контроллером в нашей дальнейшей работе мы рассмотрим вычисление оптимальных коэффициентов ПИД-регулятора для демпфирования индуцированных движением робота колебаний.

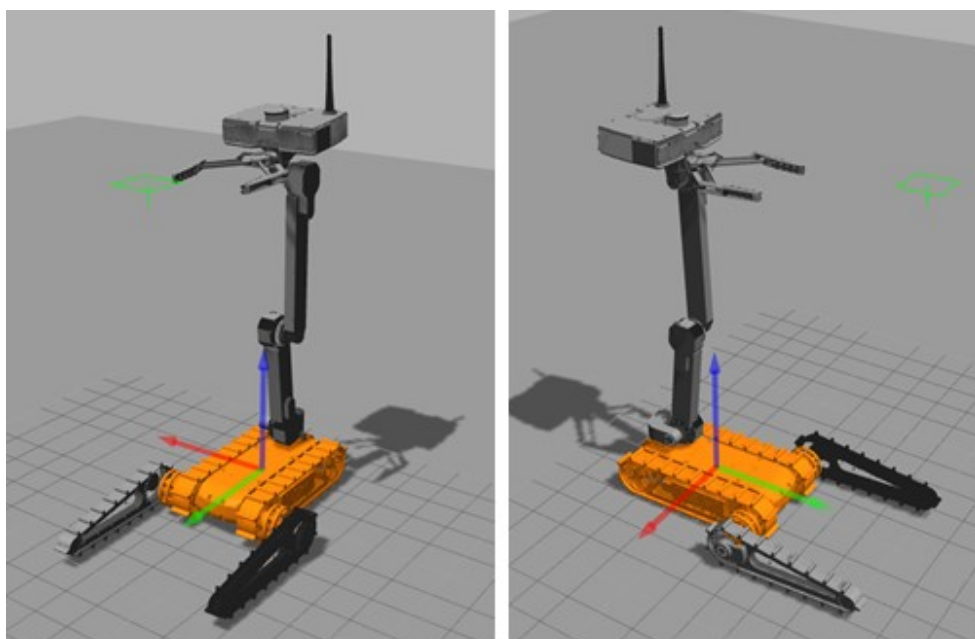


Рис. 7. Полная версия модели робота «Инженер» в Gazebo

5. Навигация в среде Gazebo.

5.1. Навигационная задача. Для проверки созданной модели использован алгоритм автономной навигации от точки к точке. Модель интегрирована в разработанную нами ранее симуляционную систему [5], которая представляет собой плоскую среду с препятствиями, где кроме автономного наземного робота присутствуют два типовых БПЛА. В рамках этой системы, два БПЛА совместно исследуют окружающую среду и выполняют картографирование. Затем полученную карту анализирует алгоритм построения пути для наземного робота. ROS пакеты `hector_quadrotor` [11] и `husky_navigation` используются для управления и навигации беспилотных летательных аппаратов и наземного робота соответственно. В качестве глобального планировщика задействован алгоритм диаграммы Вороного с алгоритмом NavFn [12], который обеспечивает быстро-интерполированное планирование пути на основе алгоритма Дейкстры. Локальный планировщик пути `DWA_planner` (Dynamic Window Approach, рус. «Подход Динамического Окна») [13, 14] использует в расчетах динамику робота и гарантирует безопасное прохождение без столкновения с препятствиями.

5.2. Симуляция и тестирование навигации. Симуляционные испытания были проведены в плоской среде; так как при отсутствии необходимости преодолевать вертикальные препятствия робот не задействует флипперы (т.е. не происходит изменения положения флипперов относительно базы), мы зафиксировали их в нулевом (горизонтальном) положении. Была использована только базовая часть робота в моделируемой среде, потому что верхняя часть робота практически не влияет на навигацию и на физическое взаимодействие с окружающей средой при перемещении по плоской поверхности. Для обхода препятствий в режиме локальной навигации робот использует лазерный дальномер - датчик ЛИДАР. Результаты навигации роботов представлены на рисунке 9.

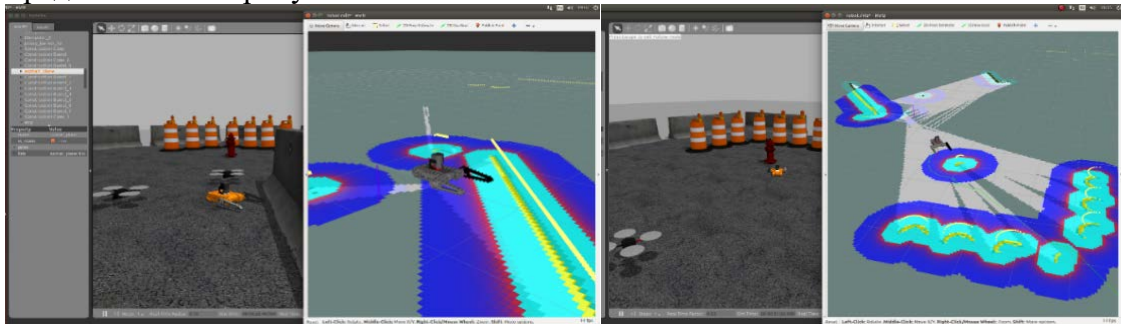


Рис. 9. Робот «Инженер» и два БПЛА в Gazebo (слева) и RViz (справа)

Заключение. В данной работе представлена первая версия модели гусеничного робота «Инженер» в среде ROS/Gazebo. На основе предоставленных компанией-производителем «Сервосила» чертежей и CAD-модели, была создана 3D-симуляция с визуализацией движения в ROS/RViz. Модель интегрирована со стекком навигации и включает в себя базу, манипулятор и захват робота. Были созданы симуляции гусеничного хода робота с помощью псевдоколес и с использованием пакета `Gazebo-tracks`. Последний метод продемонстрировал ряд критических недостатков, таких как нестабильность траков гусениц при движении, ограничение на число и радиусы направляющих роликов и отсутствие полноценного ROS API. Проведено тестирование совместной локализации и картографирования с несколькими БПЛА.

На следующих этапах нашего исследования мы планируем усовершенствовать моделирование с применением адаптивного ПИД-регулятора и расширить алгоритм планирования пути для движения по неровным участкам поверхности. Созданная модель робота «Инженер» и оригинальное программное обеспечение доступны для скачивания из открытого источника GitHub [15].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. S. Kohlbrecher, F. Kunz, D. Koert, C. Rose, P. Manns, K. Daun, and O. von Stryk, Towards Highly Reliable Autonomy for Urban Search and Rescue Robots, in *Robot Soccer World Cup* (Springer International Publishing, 2014).

2. T. Inoue, T. Shiosawa, and K. Takagi, Dynamic Analysis of Motion of Crawler-Type Remotely Operated Vehicles, in *IEEE J. of Oceanic Engineering*, 38(2), (2013), pp. 375-382.
3. E. Magid, T. Tsubouchi, E. Koyanagi, T. Yoshida, and S. Tadokoro. Controlled Balance Losing in Random Step Environment for Path Planning of a Teleoperated Crawler Type Vehicle, in *Journal of Field Robotics*, 28(6), (2011), pp.932–949.
4. R. Sheh, M.W. Kadous, C. Sammut, and Hengst, B., 2007, September. Extracting terrain features from range images for autonomous random stepfield traversal, in *IEEE International Workshop on Safety, Security and Rescue Robotics (IEEE SSRR, 2007)*, pp. 1-6.
5. Габдуллин, А., Буйвал, А., Лавренов, Р., и Магид, Е. 2016. ROS-моделирование взаимодействия между беспилотными летательными аппаратами и наземного робота для решения задачи планирования маршрута в статической среде. *III Всероссийский научно-практический семинар «Беспилотные транспортные средства с элементами искусственного интеллекта», Труды семинара.* – М: Изд-во «Перо», 2016. – 184 с. С. 21-30.
6. Shimchik, I., Sagitov, A., Afanasyev, I., Matsuno, F., and Magid, E. 2016. Golf cart prototype development and navigation simulation using ROS and Gazebo. *Int. Conf. on Mechanical, System and Control Engineering, MATEC Web Conf.*, 75, 0900.
7. Afanasyev, I.M., Sagitov, A.G., and Magid, E.A. 2016. ROS-based SLAM for a Gazebo-simulated mobile robot in image-based 3D model of indoor environment. *LNCS*, 9386.
8. Elmasry M, 2012, A tracked robot climbing stairs, www.youtube.com/watch?v=dhgcoXviDjo
9. Team “Hector”, 2013, Hector tracked vehicles common, ROS packages are available at https://github.com/tu-darmstadt-ros-pkg/hector_tracked_vehicles_common
10. A. Synodinos, Gazebo-tracks, ROS package repository (2013): <https://github.com/progtologist/gazebo-tracks>
11. ROS NavFn package, <http://wiki.ros.org/navfn>
12. ROS DWA planner, http://wiki.ros.org/dwa_local_planner
13. Fox, D., Burgard, W., and Thrun, S. 1997. The dynamic window approach to collision avoidance. *IEEE Robotics and Automation Magazine*, 4, 1.
14. Repository with ROS package for robot “Engineer”, https://github.com/Somal/Robot_Engineer
15. Model of quadrotor in ROS, http://wiki.ros.org/hector_quadrotor