

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ РОБОТОВ НА БАЗЕ ROS

Дубатовка В.В., Ярошевич Е.В.

Белорусский национальный технический университет,
Минск, Республика Беларусь

ROS (Robot Operating System) – это мета операционная система (ОС) с необходимым набором утилит (команд) для облегчения написания кода в робототехнике [1]. ROS кросс-платформенный и мультиязычный, с открытым исходным кодом. Он предоставляет такие сервисы, как аппаратная абстракция, низкоуровневое управление устройствами, реализация часто используемых операций, передача сообщений между процессами и управление пакетами [2].

ROS спроектирована как слабо связанная система, в которой процесс, называемый узлом (*node*), должен отвечать за одну задачу. Узлы общаются друг с другом, используя сообщения, проходящие через логические каналы, называемые темами (*topics*). Каждый узел может отправлять или получать данные от другого узла, используя шаблон проектирования *издатель-подписчик (publish–subscribe pattern)* [3].

Gazebo – это динамический 3D симулятор с открытым исходным кодом, который развивается Open Source Robotic Foundation и довольно тесно взаимодействует с ROS. *Gazebo* позволяет точно и эффективно моделировать роботов как в сложных условиях помещений, так и снаружи.

Таким образом, **целью** работы является разработка виртуальной робототехнической платформы, настройка среды ROS и симуляции *Gazebo*.

Материалы и методы

Тестирование проводилось на следующих робототехнических платформах: мобильный робот от Studica Robotics и летательный робот от Copter Express (COEX). С помощью средств ROS таких роботов можно быстро и гибко настроить, проводить полноценные физические симуляций в *Gazebo*. В качестве основного вычислителя используется одноплатный компьютер Raspberry Pi 4 с ОС основанной на Linux. Используется ROS Noetic, который устанавливается в виде пакетов и зависимостей ОС и позволяет в рамках одной среды наладить общение различных систем, таких как датчики расстояния, инерциальная навигационная система, драйвера моторов, интерфейсов передачи данных, являясь системой управления верхнего уровня.

Была настроена система со следующей иерархией для мобильного робота, как показано на рисунке 1. Здесь имплементированы ноды *imu* (инерциальные измерения), *odom* (одометрия), *scan* (LiDAR). Для физической симуляции *Gazebo* был построен мир с помощью внутренних инструментов и редактирования URDF-файла конфигурации.

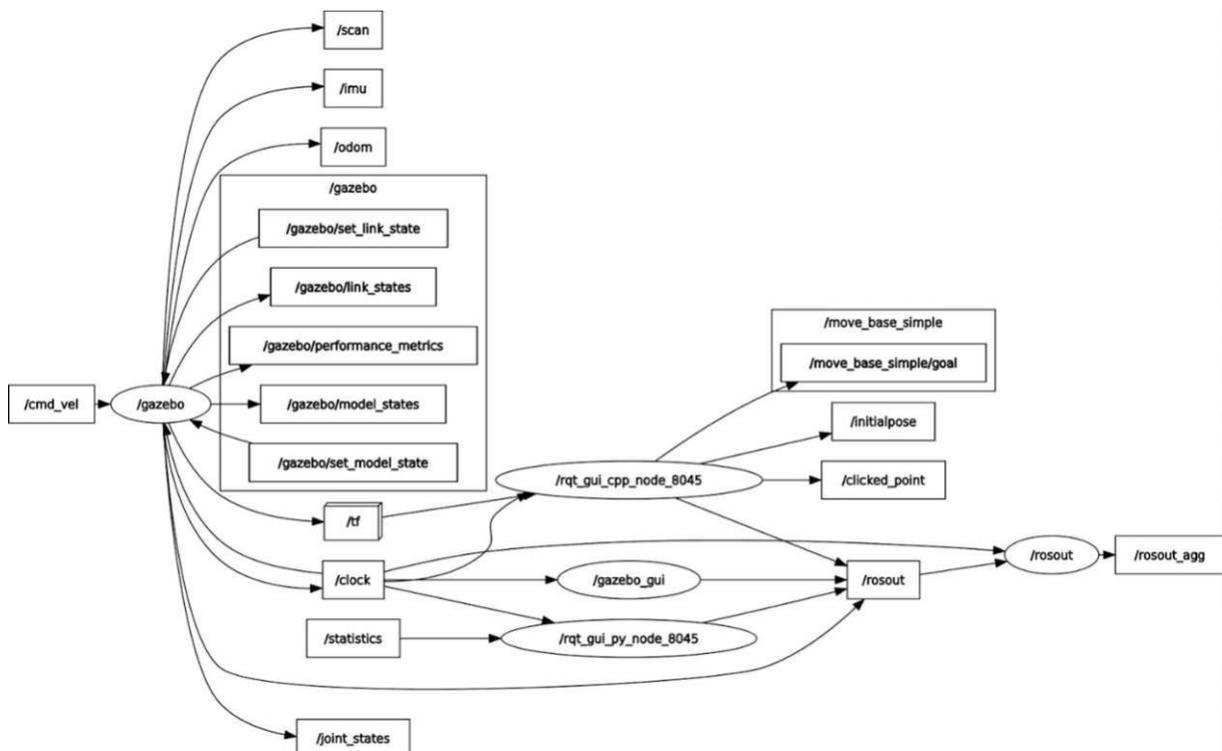


Рисунок 1 – Визуализация взаимодействия модулей ROS:RQT:Node Graph

Результаты и обсуждение

Построенная система имеет практическую ценность ввиду её гибкости и масштабируемости. В перспективе будет произведён переход на ROS2; обновлены модели роботов; написаны программы для решения более комплексных проблем используя современные методы и подходы.

Выводы

Настроена среда программирования и симуляция Gazebo.

Описаны ноды и топики для взаимодействия основных модулей робота.

Было произведено тестирование роботов в реальных условиях.

1. ROS - Robot Operating System [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://www.ros.org/>.

2. M. Ivanou, S. Mikhel and A. Maloletov, "ROS-like framework using modern development concepts and microservices," 2021 International Conference "Nonlinearity, Information and Robotics" (NIR), Innopolis, Russian Federation, 2021, pp. 1-6, doi: 10.1109/NIR52917.2021.9666141.

3. OpenAI Gym+ROS+Gazebo: обучение автономного робота [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://habr.com/ru/post/441218/>.