УДК 621.396.41

АРХИТЕКТУРНОЕ СРАВНЕНИЕ ROS1 И ROS2 В КОНТЕКСТЕ РАБОТЫ С ИНЕРЦИАЛЬНЫМИ ДАННЫМИ

Семенова Е.В. – ст. гр. ИИм-241
Кузбасский государственный технический университет имени Т.Ф. Горбачева
Россия, г. Кемерово

Долгое время в разработке автономных систем, включая робототехнику и беспилотные транспортные средства, применялась Robot Operating System первого поколения (ROS1). Однако с усложнением задач и ростом требований к отказоустойчивости и масштабируемости возникла необходимость в переходе к более современному решению — ROS2. Эта система предлагает усовершенствованные механизмы обмена сообщениями, поддержку работы в реальном времени и расширенные возможности для построения распределенных решений. В условиях, характерных для работы карьерных самосвалов, данные преимущества особенно важны — в частности, при параллельной обработке информации от нескольких лидаров и других сенсоров.

Тем не менее, несмотря на очевидные плюсы, внедрение ROS2 сопровождается определенными трудностями. Разработчики сталкиваются с проблемами при установке и настройке системы, особенно в распределенных средах с высокой нагрузкой.

Для начала рассмотрим архитектуры ROS1 и ROS2. Главное отличие между ними — переход от централизованной модели к децентрализованной: в ROS1 узлы координируются единым процессом roscore и обмениваются данными по протоколам TCPROS/UDPROS, тогда как ROS2 опирается на DDS (Data Distribution Service) и реализует взаимодействие через уровни клиентских библиотек (rclcpp, rclpy) и механизмы внутрипроцессной коммуникации. Отказ от единой точки управления в пользу распределенной системы обмена сообщениями повышает отказоустойчивость и масштабируемость, что особенно важно при передаче больших объемов инерциальных данных в условиях ограниченной пропускной способности и строгих требований к задержкам.

ROS1 — модульная и удобная для разработки платформа, долгое время служившая ориентиром для исследователей и инженеров. Но по мере роста требований к надежности, горизонтальной масштабируемости и детерминированности задержек выявились ограничения его архитектуры. Эти ограничения стали одной из причин появления ROS2, где изначально заложены принципы распределенного взаимодействия узлов, поддержка параметров качества обслуживания сообщений и улучшенные механизмы для работы в режиме реального времени. При обработке больших массивов инерциальных данных различия архитектур оказывают прямое влияние на устойчивость оценки положения, точность ориентации и надежность навигации автономных машин в сложных горных условиях ROS1 построена на централизованной архитектуре, в

которой roscore выступает в роли единого координационного узла. Обмен данными между процессами осуществляется по протоколам TCPROS и UDPROS. При работе с инерциальными датчиками (IMU) ROS1 использует стандартные сообщения sensor_msgs/Imu, обеспечивающие передачу угловых скоростей, линейных ускорений и ориентировки. Существующие пакеты (например, robot_localization) предоставляют механизмы фильтрации и слияния IMU-данных с информацией от других сенсоров. Однако отсутствие встроенной поддержки жестких временных требований и механизмов масштабируемости ограничивает применение ROS1 в системах, где требуется высокая надежность и предсказуемость задержек.

Стандартное сообщение sensor_msgs/Imu обеспечивает унифицированный формат публикации данных акселерометров и гироскопов. В ROS1 широко применяется пакет robot_localization, позволяющий выполнять фильтрацию по Калману, устранение шума, коррекцию дрейфа и объединение IMU с GNSS, одометрией или визуальными датчиками. Для интроспекции данных используется RViz, который может отображать ориентацию и динамику движения в реальном времени. Однако при высоких частотах публикации (100–1000 Гц), характерных для современных IMU, или при ограниченных вычислительных ресурсах возможно появление задержек, влияющих на точность оценки состояния.

Ключевые ограничения ROS1:

1. Централизация управления.

Зависимость от единого узла roscore создает точку отказа и ограничивает возможности горизонтального масштабирования.

2. Отсутствие QoS-параметров.

В ROS1 невозможно настраивать политику доставки сообщений (надежность, задержки, управление очередями). При работе с высокочастотными IMU-данными это критично: потеря даже части сообщений или вариативность задержки ухудшает результаты фильтрации и приводит к нестабильной оценке положения.

3. Ограничения реального времени.

Архитектура ROS1 изначально не ориентирована на жесткие временные требования. Это может вызывать непредсказуемые задержки в приеме и обработке IMU-данных, которые играют ключевую роль в системах навигации и стабилизации.

4. Накладные расходы при преобразовании данных.

Частые сериализация/десериализация сообщений и преобразования между различными форматами усложняют использование IMU в контурах управления, требующих минимальной задержки.

5. Сокращение поддержки.

Сообщество активно мигрирует на ROS2, поэтому развитие ROS1 замедляется: обновления, исправления и улучшения выходят реже, что увеличивает технический долг проектов.

В ROS2 полностью отказались от концепции центрального узла (roscore), характерной для ROS1, и перешли на стандарт DDS. Это устраняет единую точку отказа и позволяет каждому узлу автономно обнаруживать и связываться с другими участниками системы, что особенно важно в распределенных навигационных комплексах, включающих несколько IMU, GNSS-приемников и вычислительных модулей. DDS предоставляет тонкую настройку параметров качества обслуживания (QoS), включая надежность доставки, глубину очереди, политику истории сообщений и гарантии задержек. Благодаря этому можно обеспечить детерминированную передачу инерциальных данных с минимальной вариативностью задержек — критически важное условие для устойчивой ориентации, точной оценки положения и корректной работы навигационных алгоритмов в сложном горном рельефе.

В экосистеме ROS2 существуют специализированные решения для обработки данных IMU, включая пакеты из составов Autoware и robot_localization, которые реализуют модульную архитектуру этапов фильтрации и коррекции инерциальной информации. Общая схема интеграции IMU, LiDAR и камер в единую модель окружения строится по принципам модульного сенсорного фьюжна в реальном времени. Рекомендуемый конвейер предобработки IMU-данных включает:

1. фильтрацию шумов высокочастотных компонентов (noise filtering);
2. корректировку дрейфа и смещения нуля (bias correction);
3. сглаживание и подавление вибраций (vibration compensation);
4. синхронизацию временных меток с другими сенсорами (timestamp alignment);
5. преобразование данных IMU в единую систему координат (frame transformation);
6. слияние IMU с GNSS и одометрией в рамках единого фильтра состояния (multi-sensor state estimation).

Такой подход обеспечивает воспроизводимость и повторное использование отдельных модулей при изменении аппаратной конфигурации самосвала, параметров навигационной системы или требований к точности ориентации.

ROS2 широко применяется совместно с современными алгоритмами инерци-ально-визуальной и инерциально-LiDAR локализации, такими как LIO-SAM, которые обеспечивают высокоточную оценку траектории и построение карт в режиме реального времени. Благодаря встроенным механизмам DDS, данные IMU доставляются в SLAM-узлы с минимальной вариативностью задержки даже при высокой нагрузке вычислительной системы.

Среди ключевых преимуществ ROS2 можно выделить следующие:

1. Масштабируемость.

Архитектура ROS2 позволяет эффективно работать с большим числом сенсоров и вычислительных узлов, включая высокочастотные IMU, GNSS-приемники и другие навигационные источники. Система сохраняет производительность даже при росте частоты публикации данных и усложнении задач интеграции сенсоров.

2. Надежность.

Продвинутые механизмы взаимодействия между компонентами, а также поддержка контроля качества обслуживания (QoS), обеспечивают устойчивую работу системы в условиях вибраций, механических воздействий, ограниченной пропускной способности сети и высоких требований к детерминированности передачи IMU-данных.

3. Гибкость.

Платформа предоставляет широкие возможности для внедрения и замены алгоритмов обработки инерциальной информации — фильтров ориентации, моделей слияния данных, модулей оценки состояния. Это важно при необходимости оперативной адаптации системы под новые датчики или условия эксплуатации без существенных изменений в архитектуре.

4. Совместимость.

ROS2 поддерживает взаимодействие с широким спектром аппаратных и программных компонентов — от IMU и GNSS до локализационных модулей и алгоритмов контроля движения. Это упрощает интеграцию навигационных подсистем и расширение функциональности.

Несмотря на существенные достоинства, использование ROS2 связано и с рядом сложностей.

1. Сложности с DDS и настройкой QoS

Одной из наиболее заметных проблем является работа с DDS, на которой основана коммуникационная модель ROS2. Хотя DDS обеспечивает необходимые гибкость и масштабируемость, настройка параметров качества обслуживания (QoS) может быть нетривиальной.

2. Ограничения и задержки при передаче высокочастотных IMU-данных между машинами

При использовании ROS2 для передачи IMU-данных между несколькими вычилительными модулями могут возникнуть следующие проблемы: потери данных и высокая нагрузка на сеть.

3. Ограниченная поддержка некоторых полей в сообщениях sensor_msgs/Imu

Для некоторых сенсорных модулей наблюдаются ограничения в поддержке дополнительных полей IMU-сообщений. Пользователи отмечают случаи отсутствия расшренных метаданных (таких как температура датчика или точность измерений), даже если оборудование предоставляет подобную информацию.

4. Проблемы с производительностью при обработке больших потоков данных

Высокочастотные IMU-потоки могут создавать повышенную нагрузку на процессоры и систему обмена сообщениями. Для повышения эффективности рекомендуется использовать композицию узлов, размещенных в одном процессе, однако это усложняет архитектуру системы и повышает требования к проектированию приложения.

5. Зависимость от реализации DDS

Производительность и надежность ROS2 существенно зависят от выбранной реализации DDS. Разные реализации демонстрируют различную латентность, устойчивость к перегрузкам и эффективность доставки сообщений, что требует тщательных испытаний при разработке навигационных подсистем.

6. Ограниченная поддержка и документация отдельных функций

Некоторые функциональные возможности ROS2, включая новые типы QoS, отдельные средства синхронизации и параметры реального времени, имеют ограниченную документацию. Это может затруднять внедрение современных методов обработки и слияния инерциальных данных в промышленных условиях.

Заключение

Таким образом, переход от централизованной архитектуры ROS1 к децентрализованной модели ROS2 на основе DDS открывает новые возможности для построения надежных и масштабируемых систем обработки LiDAR-данных. Вместо одного узла координации теперь узлы обмениваются сообщениями напрямую, что снижает вероятность полного отказа системы и упрощает параллельную работу множества сенсоров и вычислительных модулей.

Вместе с тем внедрение ROS2 налагает свои требования: настройка различных реализаций DDS, тонкая корректировка политик QoS и обеспечение стабильной передачи крупных объемов точечных облаков требуют внимания и опыта. На практике это означает необходимость тщательно продуманного плана миграции, поэтапной проверки компонентов и регулярного мониторинга сетевой и вычислительной нагрузки.

Список литературы:

1. Дубинкин Д.М. Создание алгоритма обработки данных систем беспилотного управления карьерного самосвала для построения высокоточных 3D-карт местности / Д.М. Дубинкин, В.Ю. Садовец, И.С. Сыркин, Е.В. Коршунова // Уголь. – 2024. – № S11(1187). – С. 116–122. – DOI: 10.18796/0041-5790-2024-11S-116-122.

2. Оптимизация показателей работы роботизированного экскаваторно-автомобильного комплекса разреза / Ю. Е. Воронов, А. Ю. Воронов, Д. М. Дубинкин, О. С. Максимова // Уголь. – 2024. – № 7(1182). – С. 62-67. – DOI 10.18796/0041-5790-2024-7-62-67.

3. Анализ показаний телеметрии при эксплуатации карьерных самосвалов грузподъемностью 220 тонн / А. И. Бокарев, В. А. Дианов, А. Б. Карташов [и др.] // Горное оборудование и электромеханика. – 2024. – № 5(175). – С. 52-60. – DOI 10.26730/1816-4528-2024-5-52-60.

4. ROS on DDS [Электронный ресурс] / ROS 2 Design Working Group. – 2023. – URL: https://design.ros2.org/articles/ros_on_dds.html (дата обращения: 21.05.2025).

5. Addison J. ROS 2 Architecture Overview [Электронный ресурс] / J. Addison // Au-tomatic Addison. – 2023. – URL: https://automaticaddison.com/ros-2-architecture-overview/ (дата обращения: 22.05.2025).