УДК 004.4

КОНТЕЙНЕРИЗАЦИЯ AUTOWARE С УДАЛЁННЫМ СИМУЛЯТОРОМ CARLA, РАЗВЁРНУТЫМ НА ОТДЕЛЬНОМ СЕРВЕРЕ, С ИСПОЛЬЗОВАНИЕМ ПАКЕТА CARLA-AUTOWARE-BRIDGE

Мильков А.В., студент гр. ИТб-212, IV курс
Научный руководитель: Сыркин И.С., к.т.н., доцент
Кузбасский государственный технический университет
имени Т.Ф. Горбачева,
Россия, г. Кемерово

Современные системы автономного вождения требуют проведения большого количества испытаний в контролируемых условиях. Полноценные полигонные испытания связаны с высокими затратами и рисками, поэтому всё большую роль играют программные симуляторы дорожной обстановки, такие как CARLA, а также специализированные программные стеки, например Autoware. Их совместное использование позволяет воспроизводить сложные транспортные сценарии и отрабатывать алгоритмы управления транспортным средством без участия реальной техники.

При этом в реальных проектах симулятор и стек автономного вождения нередко развёртываются на разных вычислительных узлах: симулятор - на выделенном графическом сервере, а Autoware - в контейнере на отдельном сервере разработчика или в облачной инфраструктуре. Это требует решения ряда задач, связанных с контейнеризацией, сетевой маршрутизацией и настройкой средств обмена данными.

Цель данной работы - описать архитектуру и практическую реализацию распределённого стенда, в котором Autoware запускается в контейнере Docker на одном сервере, а симулятор CARLA - на другом, при этом связующим звеном выступает пакет carla-autoware-bridge.

Autoware - это открытый программный стек для автономного вождения, построенный на базе ROS2. Он включает подсистемы локализации, восприятия, планирования движения и управления. Использование ROS2 обеспечивает модульность системы и удобный обмен сообщениями между компонентами.

CARLA представляет собой симулятор дорожной обстановки с поддержкой физически корректной модели транспортных средств, реалистичных датчиков (лидар, камеры, радары) и гибкой системой сценариев и предоставляет API для удалённого управления и получения данных, что делает возможным её интеграцию с внешними системами.

Пакет carla-autoware-bridge выполняет роль моста между симулятором и Autoware. Он подключается к CARLA по сети, получает синтетические данные с датчиков, преобразует их в сообщения ROS2 и передаёт в соответствующие топики. В обратную сторону через мост отправляются команды продольного и

поперечного управления (газ, тормоз, рулевое управление), которые затем применяются к виртуальному автомобилю в симуляторе.

Контейнеризация Autoware позволяет изолировать программную среду, упростить управление зависимостями и обеспечить воспроизводимость развёртывания на разных вычислительных узлах.

Для него доступны готовые Docker-образы, включающие необходимый набор пакетов ROS2, драйверов и инструментов. В типичном сценарии развёртывания выполняются следующие шаги:

- Выбор базового образа. В качестве основы используется официальный образ Autoware для соответствующей версии ROS2 и операционной системы (например, образ на базе Ubuntu с установленным ROS2 Humble и autoware-universe).
- Настройка окружения. В образ добавляются переменные окружения, необходимые для работы ROS2 и DDS, в частности RMW_IMPLEMENTATION=rmw_cyclonedds_cpp или другая реализация, выбранная для проекта.
- Подключение GPU и X-сервера. При необходимости визуализации (RViz) контейнер запускается с доступом к графическому серверу хоста и видеокарте (опции --gpus all, --env DISPLAY, проброс сокета X11).
- Организация рабочих пространств. Внутри контейнера создаётся рабочее пространство ROS2, в которое помещаются пакеты, относящиеся к проекту (конфигурации транспортного средства, карты, собственные узлы). Для удобства разработки рабочее пространство может быть примонтировано с хоста через --volume, что позволяет редактировать код вне контейнера.
- Использование режима host-network. Для упрощения сетевого взаимодействия контейнер часто запускается с параметром --network host, что позволяет использовать IP-адрес и порты хостовой машины и упрощает интеграцию с удалёнными сервисами, такими как CARLA.

Благодаря контейнеризации разработчик получает унифицированную среду для работы с Autoware, независимую от конфигурации конкретного сервера и исключающую конфликты библиотек и версий пакетов.

Архитектура распределённой системы с удалённым симулятором включает два физических сервера:

- Сервер симулятора - на нём развёрнут CARLA, использующий графический ускоритель и локальные ресурсы.
- Сервер - на нём в контейнере Docker запускаются Autoware и пакет carla-autoware-bridge.

Связь между компонентами осуществляется по сети. CARLA выступает в роли сервера, ожидающего подключения клиентов по заданному TCP-порту. Контейнер с Autoware и мостом выступает в роли клиента и подключается к удалённому серверу, используя его IP-адрес и порт, переданные в параметрах запуска моста или в переменных окружения.

Ключевые аспекты такой архитектуры:

- Сетевая доступность. Порт CARLA должен быть открыт на сервере симулятора и доступен с сервера Autoware. При наличии межсетевых экранов необходима настройка правил проброса трафика.
- Задержки и пропускная способность. Межсерверное соединение должно обеспечивать достаточно низкие задержки и пропускную способность для передачи облаков точек, изображений и других данных в режиме реального времени.
- Настройки DDS. ROS2 использует DDS для обмена сообщениями между узлами. При распределённой конфигурации важно корректно настроить параметры участника DDS (адреса интерфейсов, метаданные), чтобы все узлы Autoware в контейнере могли обмениваться сообщениями без конфликтов и временных задержек.
- Безопасность. При использовании внешних сетей рекомендуется ограничивать доступ к портам симулятора и контейнера с Autoware, а также рассматривать использование защищённых туннелей.

Практическая схема развёртывания включает несколько последовательных этапов

На сервере, оснащённом GPU, устанавливается или разворачивается в контейнере симулятор CARLA нужной версии. При запуске задаётся порт для RPC-подключений и, при необходимости, параметры карты (Town10, пользовательские сцены и т.д.). Важным условием является доступность порта симулятора из внешней сети.

На втором сервере разворачивается Docker-образ Autoware. Контейнер запускается с параметрами:

- --network host для минимизации проблем с маршрутизацией;
- при необходимости --gpus all и пробросом X-сервера;
- примонтированными каталогами для карт, конфигураций и логов.

Внутри контейнера активируется рабочее пространство ROS2, содержащее конфигурации транспортного средства, карт и необходимые пакеты.

Пакет carla-autoware-bridge устанавливается либо в том же рабочем пространстве, либо используется уже включённый в образ. При запуске моста указываются:

- IP-адрес и порт удалённого сервера CARLA;
- используемая карта и параметры сцены;
- топики, по которым мост будет публиковать данные сенсоров (лидар, камеры, IMU) и принимать команды управления от Autoware

Мост подключается к CARLA, начинает получать синтетические данные, преобразует их в сообщения ROS2 и передаёт в соответствующие узлы Autoware. Параллельно ему на основе этих данных выполняет локализацию, восприятие и планирование, формирует управляющие воздействия и отправляет их обратно в симулятор через мост.

Для контроля работы распределённого стенда используются инструменты:

- средства визуализации ROS2 (например, RViz) для проверки корректности облаков точек, карт и траекторий;
- просмотр логов контейнера Docker и узлов ROS2;
- сетевые утилиты для контроля задержек и нагрузки на канал связи.

При необходимости изменяются параметры частоты публикации данных, разрешений сенсоров и конфигурация DDS для достижения стабильной работы при наличии сетевых задержек.

Использование контейнеризации Autoware с удалённым симулятором CARLA даёт ряд существенных преимуществ:

- Гибкость инфраструктуры. Симулятор и стек автономного вождения могут размещаться на разных физических серверах в зависимости от доступных ресурсов.
- Масштабируемость. На одном сервере можно запускать несколько экземпляров Autoware в отдельных контейнерах, подключая их к одному или нескольким симуляторам для параллельной отработки различных сценариев.
- Повторяемость экспериментов. Стандартизированный Docker-образ позволяет легко переносить экспериментальную среду между разработчиками и серверами без сложной настройки.
- Изоляция и безопасность. Контейнеры ограничивают влияние тестируемых компонентов на базовую систему и друг на друга.

Перспективными направлениями развития являются использование более гибких транспортных уровней и шлюзов данных, интеграция с системами распределённого логирования и автоматизированного развёртывания (CI/CD), а также расширение набора сценариев для тестирования алгоритмов автономного вождения в условиях, максимально приближённых к реальным.

В заключение можно сказать, что реализованный подход позволяет построить распределённый стенд, в котором симулятор CARLA, работающий на удалённом сервере, и стек Autoware, развёрнутый в контейнере Docker, интегрируются посредством пакета carla-autoware-bridge в единую испытательную среду. Такая архитектура обеспечивает воспроизводимость вычислительных экспериментов, упрощает сопровождение программной инфраструктуры и даёт возможность гибко масштабировать вычислительные ресурсы при исследовании и отладке алгоритмов автономного вождения.

Список источников:

1. Docker Documentation - Текст: электронный // docker.com – URL: https://docs.docker.com/

2. ROS2 Documentation - Текст: электронный // docs.ros.org – URL: https://docs.ros.org/en/ros2_documentation/

3. Autoware Foundation. Autoware Documentation - Текст: электронный. – URL: https://autoware.org

4. CARLA Autonomous Driving Simulator - Текст: электронный. – URL: https://carla.org

5. TUM Autonomous Driving. Carla-Autoware-Bridge Project - Текст: электронный. – URL: https://github.com/TUMFTM/Carla-Autoware-Bridge/tree/main?tab=readme-ov-file Kuratorov A.V., "Containerization of ROS-based Robotic Applications", Journal of Robotics, 2022.