

ПРОГРАММИРОВАНИЕ РОБОТОВ

Р. С. Катмышков, Н. В. Мисько

*Гродненский Государственный Университет им. Я. Купалы. г. Гродно
Научный руководитель Е. А. Сетько, кандидат физ.-мат наук, доцент*

Робототехника играет важную роль в современном мире и широко применяется в промышленности, медицине, ВПК, агропромышленном комплексе. В настоящее время основными проблемами робототехники являются:

- этика и законодательство,
- работа и замещение рабочей силы.

Развитие робототехники поднимает вопросы о правовых и этических аспектах использования роботов. Как определить ответственность за действия автономных роботов, особенно если они причиняют вред людям или имуществу? Какие ограничения следует установить на создание и использование роботов в военных или медицинских целях?

Установление чёткой ответственности за действия автономных роботов является сложным вопросом. Анализ случаев использования роботов в различных областях выявляет, что ответственность часто размывается между производителями, программистами и конечными пользователями [1]. Существуют три закона робототехники, сформулированных в научной фантастике [2]:

1. Робот не может причинить вред человеку или своим бездействием допустить чтобы человеку был причинен вред.

2. Робот должен повиноваться всем приказам, которые даёт человек, кроме тех случаев, когда эти приказы противоречат первому закону.

3. Робот должен заботиться о своей безопасности в той мере, в которой это не противоречит первому или второму законам

В настоящее время активно производится замена человеческой рабочей силы роботами: в американских школах уже есть роботы-охранники, создан человекоподобный робот-лётчик, роботы начали дирижировать оркестрами.

Чтобы роботы органично вписались в нашу жизнь необходимо создание современного и качественного программного обеспечения. Существуют различные системы программирования роботов и один из самых используемых и гибких - открытый фреймворк для написания программного обеспечения, работающий как мета-операционная система на базе ОС Linux [3]. Это Robot Operating System (ROS). ROS представляет собой набор из различных широко (и не очень) известных библиотек, таких как:

OpenCV — библиотека, содержащая алгоритмы компьютерного зрения и обработки изображений;

PCL- библиотека для работы с облаками 3D-точек;

Ogre — объектно-ориентированный графический движок с открытым исходным кодом;

Orocos — библиотека для управления роботами (например, расчёт кинематики).

Также в ROS входят драйвера для различных манипуляторов и сенсоров (включая MS Kinect). Но что же отличает ROS от простой сборки библиотек? Основополагающим преимуществом является клиент-серверная архитектура ROS — разработчики реализовали механизм пересылки сообщений между различными объектами, возможность построения распределенных систем, предоставление brige'ей к языкам C++ и Python.

В разных областях робототехники используют ROS в качестве платформы разработки по определенным причинам. Исследования в области искусственного интеллекта и машинного обучения имеют свои преимущества при его использовании, так как ROS предоставляет среду для обучения роботов в виртуальных средах или на реальных устройствах. Однако существуют и ограничения: необходимость в интеграции машинного обучения может создавать сложности при разработке и отладке, особенно при работе с реальными роботами, где требуется учет физических ограничений и динамики.

Использование ROS позволяет управлять множеством промышленных роботов, интегрируя различные датчики и устройства, оптимизируя процессы производства и автоматизируя задачи. Однако и тут есть свои ограничения, ведь в промышленности также важны вопросы безопасности и отказоустойчивости. Некоторые промышленные системы имеют собственные протоколы и стандарты, что может создавать сложности при интеграции с ROS.

ROS остаётся крайне популярным инструментом в разных областях со своей гибкостью и функциональностью, но для успешного применения в каждой сфере необходимо учитывать специфические требования и особенности этой области.

У ROS есть альтернативы в разработке программного обеспечения. Такие как Robotnik Framework , Microsoft Robotics Developer Studio (MRDS) , YARP (Yet Another Robot Platform) , Gazebo и т. д. В зависимости от конкретных потребностей проекта, другие платформы могут быть более подходящими, учитывая их особенности и ограничения. Но ROS, безусловно, имеет преимущества в виде широкой экосистемы пакетов, большого сообщества разработчиков, гибкости и масштабируемости.

На сегодняшний день, ROS является востребованной и широко используемой платформой для разработки программного обеспечения в робототехнике. Современные роботы оснащённые эффективным программным обеспечением становятся реальностью нашей жизни и решают многие вопросы:

- **Эффективность и производительность.** Роботы и автоматизированные системы часто способны выполнять определённые задачи более быстро и точно, чем человек. Это может повысить общую эффективность производства и снизить затраты.

- **Экономия затрат.** Инвестиции в роботизацию могут снизить операционные расходы компаний в долгосрочной перспективе. В частности, в некоторых отраслях, где трудозатраты высоки, использование роботов может быть экономически оправданным.

● Точность и надёжность. Роботы могут быть запрограммированы для выполнения задач с высокой точностью и надёжностью, что особенно важно в сферах, где допуски ошибок минимальны, например, в производстве или медицинской диагностике.

● Автоматизация рутинных задач. Рутинные и монотонные задачи могут быть легко автоматизированы, что позволяет человеку сосредотачиваться на более творческих и сложных аспектах работы

● Технологический прогресс. Развитие технологий, таких как искусственный интеллект, машинное обучение и робототехника, делает возможным автоматизацию более широкого спектра задач.

Не смотря на имеющиеся проблемы, использование робототехники - это шаг в будущее и данное направление будет постоянно развиваться.

Список литературы

. Проблемы современной робототехники: какие есть и как решают? [Электронный ресурс]. — Режим доступа: <https://robot-ex.ru/ru/article/problemi-sovremennoy-robototekhniki-kakie-est-i-kak-reshayut-66842> . — Дата доступа:

2. Азимов А.Хоровод.[Электронный ресурс] - Режим доступа: <http://asimomvonline.ru/short-stories/khorovod/read/?page=3> — Дата доступа:

3. Основы работы с Robotic Operating System. [Электронный ресурс] - Режим доступа: <https://habr.com/ru/articles/128024/> - Дата доступа: 18.11.2023.

УДК621.3

РОБОТИЗИРОВАННАЯ ПНЕВМАТИЧЕСКАЯ СТАНЦИЯ МЕХАТРОНИКИ

Ю. Н. Матрунчик, И. С. Марченко, К. А. Беликова, И. И. Войтеховский
Белорусский национальный технический университет,
УО «Национальный детский технопарк»

Мехатроника является исключительно динамично продвигающимся направлением современной науки, техники и технологии, которое определяет облик современной техносферы. Основная задача мехатроники состоит в создании интеллектуальных машин и движущихся систем, обладающих качественно новым функционалом.

Целью данной работы является проблема роботизации производственного процесса транспортирования, базирования, ориентации и сортировки деталей.

Задачи работы – провести анализ существующих аналогов мехатронных станций сортировки и транспортировки деталей; разработать пневматическую