

УДК 004.896+004.512.4

ИССЛЕДОВАНИЕ ВЗАИМОДЕЙСТВИЯ РОБОТОВ-МАНИПУЛЯТОРОВ С ОКРУЖАЮЩЕЙ СРЕДОЙ

В.В. Рябченко, Н.Н. Дацун

Донецкий национальный технический университет
кафедра прикладной математики и информатики

E-mail: VladRia@mail.ru

Аннотация

Рябченко В.В., Дацун Н.Н. Исследование взаимодействия роботов-манипуляторов с окружающей средой. Статья посвящена проблемам разработки программного обеспечения для управления роботами, моделирование их поведения, реакций на внешние воздействия. Описан алгоритм обработки сигналов внешней среды, реализация поведения, приведены результаты лабораторных экспериментов.

Введение

Непрерывно развиваясь, промышленная робототехника ведет свою историю с 60-х гг. XX века, оказывая сильное положительное влияние на качество и интенсивность практически любого производства. Робот-манипулятор – это автономное устройство, состоящее из механического манипулятора и перепрограммируемой системы управления, которое применяется для перемещения объектов в пространстве и для выполнения различных производственных процессов. Роботы-манипуляторы широко используются в лабораторных исследованиях, автоматизированном промышленном производстве, измерениях и т.д. [1-3].

Объектом исследования является робот-манипулятор Katana фирмы Neuronics AG (Швейцария) [1]. Среди роботов-манипуляторов продукция фирмы Neuronics выделяется такими преимуществами как относительно доступная цена, интеллектуальность, наличие целого семейства роботов Katana для использования в различных целях.

Цели работы

Область программирования роботов обладает собственным кругом задач, которые должны иметь решение как в практической сфере, так и в сфере теоретической. Среди прочих проблем основными являются: вопросы взаимодействия робота и человека во всем многообразии способов и целей, искусственного интеллекта, деятельности робота в пространстве, моделирования поведения робота и т.д.

В данной работе преследуются следующие цели:

- исследование методов программного управления роботами-манипуляторами Katana фирмы Neuronics в операционных системах Linux и MS Windows;
- разработка программного интерфейса на основе библиотеки KNI [1] и микропрограммного обеспечения (Firmware) робота;
- моделирование поведения робота: обработки сигналов внешнего мира таким образом, чтобы получить ожидаемые реакции робота;
- реализация управления роботом по усилию (внешнему силовому воздействию).

Данная работа выполнялась в рамках украинско-французского исследовательского проекта ДонНТУ - Université de Cergy-Pontoise и НИР 11-316 «Біо-подібні моделі гуманідних роботів у ритмічній взаємодії з їх навколишнім оточенням» (под рук. Борисенко В.Ф., Хоменко В.Н., Мельника А.А.) [2].

Разработка программного интерфейса на основе KNI

Исследуемый робот Katana 5M180 (рис. 1) размещен в лаборатории кафедры ЭАПУ ДонНТУ.



Рисунок 1 - Katana 5M180 ф. Neuronics AG

Katana управляется контроллером, который предоставляет интерфейс – предустановленное множество команд (Firmware). Этот подчиненный компонент системы управления позволяет роботу следовать командам и оповещает ведущий компонент (Software) о состоянии робота. В свою очередь, ведущий компонент – это программа, посылающая поток команд в соответствии с желаемым поведением робота. Таким образом, архитектура разработанного программного обеспечения для управления роботом разделена на две части: Software – верхняя часть, выполняющаяся на компьютере, и нижняя – Firmware, которая выполняется процессором робота. Software выполняется на компьютере, Firmware – процессором робота, а пакеты между ними (команды, ответы) передаются через COM-порт.

Управление роботом только с помощью команд контроллера несет в себе значительные трудности, такие как: необходимость формирования каждого пакета для установления соединения, движения, получения данных, анализ ответов. В то же время, объект управления может быть в полной мере представлен на различных уровнях абстракции. Поэтому были разработаны библиотеки, которые предоставляют программный интерфейс (классы, функции, сущности, и т.д.) для робота Katana, которые могут быть использованы для решения широкого круга научных и практических задач. Наш программный интерфейс основан на KNI (Katana Native Interface) – C++ библиотеке управления линейкой роботов Katana, распространяемой их производителем – Neuronics AG.

Архитектура интерфейса разделена на пять уровней (рис. 2):

1. На коммуникационном уровне (KNI Communication Layer) реализованы функции отправки, получения и представления управляющей информации передаваемой между компьютером и роботом. Он состоит из:
 - 1.1. Уровня устройств (KNI Device Layer) где описываются сущности (структуры, функции) передачи данных.
 - 1.2. Уровня протокола (KNI Protocol Layer), на котором реализованы эти сущности в соответствии с текущими техническими особенностями (операционная система, типы устройств, портов и т.д.). Так как мы использовали Linux в качестве ОС и COM-порт для передачи данных, были использованы соответствующий инструментарий (termios.h, и т.д.)
2. На базовом уровне описаны такие сущности как: робот Katana, мотор, захват, сенсор, и т.д. и реализованы базовые функции (повернуть i-е сочленение в позицию *XXenc*,

получить текущие показания сенсоров, двигаться в позицию “XX, XX, XX, XX, XX”, и т.д.). Классы и функции на этом уровне используют сущности коммуникационного уровня и не зависят от специфики их реализации.

3. Сущности абстрактного уровня используются для решения проблем прямой и обратной кинематики, систем координат и предоставляют интеллектуальные, но простые в использовании функции для управления роботом Katana.
4. На уровне приложений программист может воплотить любое желаемое поведение робота, используя структуры и функции из любого уровня программного интерфейса.

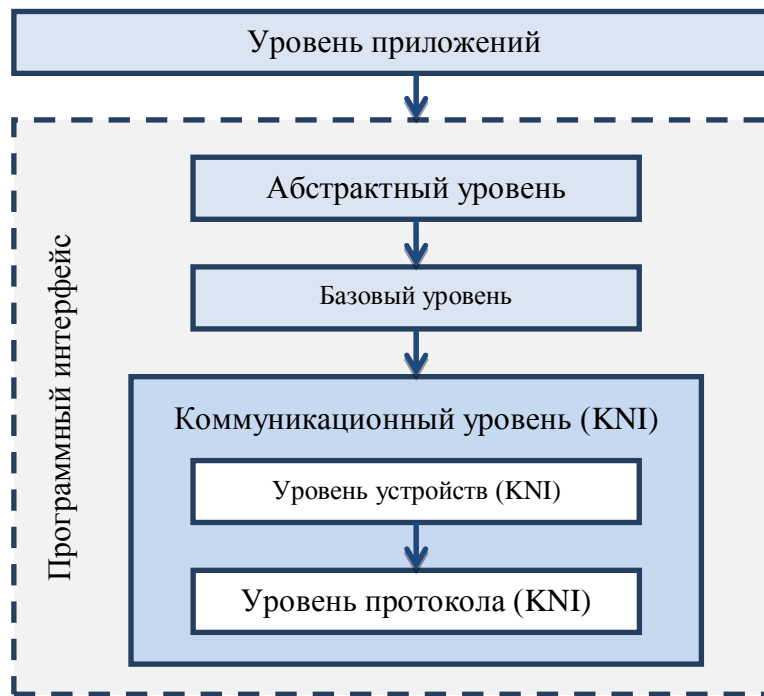


Рисунок 2 - Архитектура программного интерфейса

Реализация реакции робота на внешнее воздействие

Для цифровой интерпретации физического воздействия мы использовали:

- Датчики деформации;
- Аналого-цифровой конвертер – National Instruments PCI 6023E;
- Comedi – коллекция драйверов для получения данных от встраиваемых карт и датчиков. Comedilib – пользовательская библиотека, которая предоставляет программный интерфейс для устройств получения аналоговой информации.

Совокупность этих инструментов позволила получать цифровой эквивалент силы, приложенной ко второму звену робота. Другими словами, в то время, когда робот не испытывает внешнего воздействия, датчики показывают некое значение, которое принимается как значение покоя. Если к звену приложить силу (попытаться сдвинуть его), то значение датчиков отклонится от значения покоя пропорционально приложенной силе. Если воздействовать на звено с противоположной стороны (изменить вектор силы), значение будет отклоняться в другую сторону.

Поведение робота, которое мы реализовали, заключается в следующем: он будет следовать за человеком, который его направляет (в то время как сдвинуть включенный робот невозможно). Это можно легко представить как ритмическое взаимодействие, такое как рукопожатие. Робот не будет противиться внешнему воздействию, а даже следовать ему. Алгоритм реализации поведения робота Katana (рис. 3)

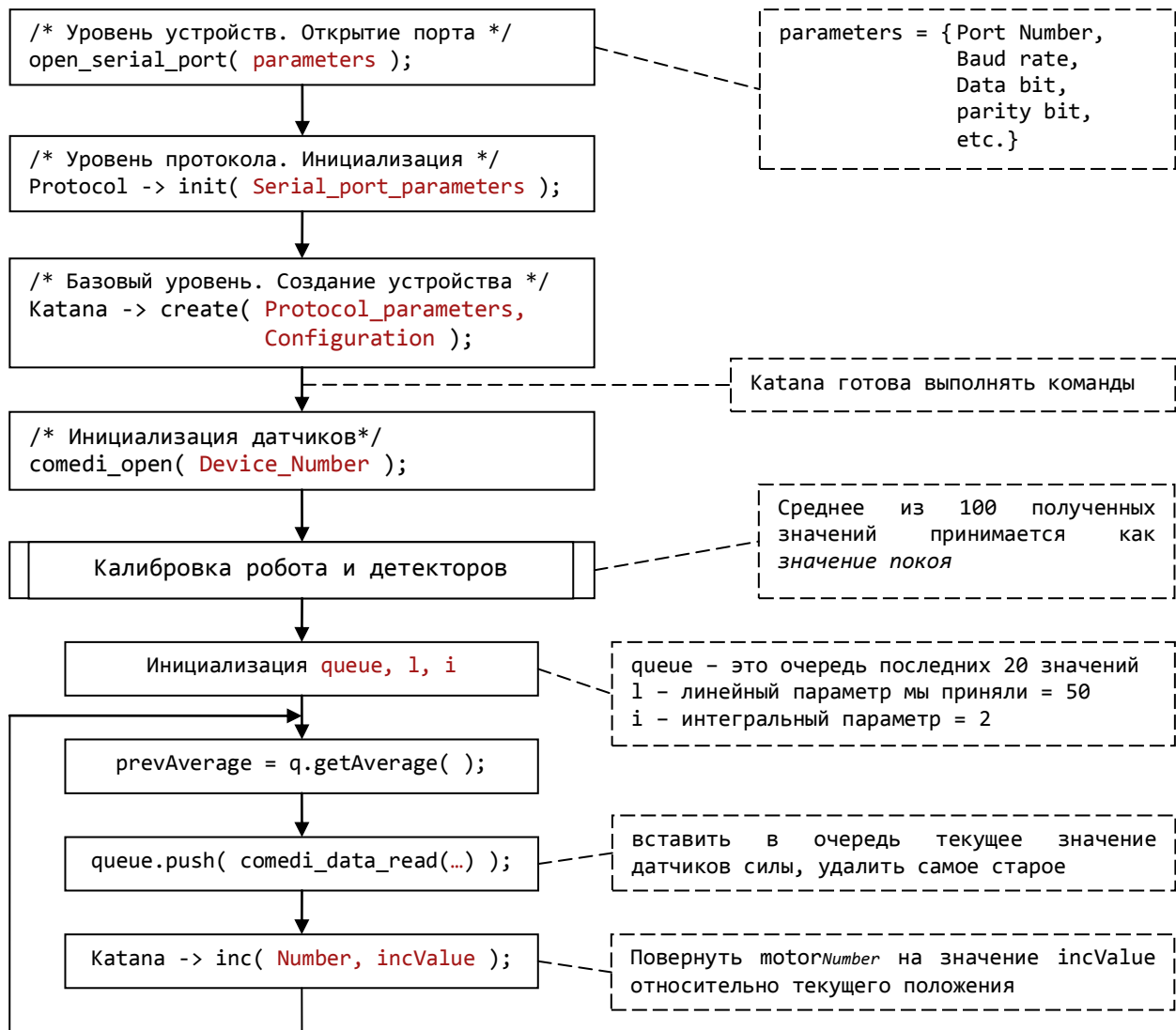


Рисунок 3 - Алгоритм обработки сигналов

Объект *queue* содержит 20 самых последних значений датчиков. Метод *Queue::getAverage()* возвращает среднее этих значений. Метод *Queue::pop(int recentValue)* удаляет самое старое значение очереди и вставляет *recentValue* как последнее.

При вычислении величины поворота двигателя перемещающего сочленение, к которому приложена сила, используется не текущее показание датчиков силы, а среднее последних 20. Такой подход позволяет свести к минимуму так называемый шум – неконтролируемое отклонение показаний.

Вычисление величины *incValue* на которую нужно повернуть двигатель:

$$\begin{aligned}
 incValue &= l \cdot \text{Среднее}_{20} + i \cdot \frac{\text{Среднее}_{20} - \text{Пред.Среднее}_{20}}{t_{\text{сейчас}} - t_{\text{пред.}}} = \\
 &= l \cdot AF + i \cdot AF' = l \cdot AF + i \cdot \frac{\Delta AF}{\Delta t} = l \cdot (\text{линейное выражение AF}) + i \cdot (\text{интегральное выражение AF}) \quad (1)
 \end{aligned}$$

Результаты эксперимента

На рисунке 4 приведен график, показывающий изменения во времени приложенной силы и положения звена при ритмическом взаимодействии с человеком.

Как видно из графика, звено робота перемещается в соответствии с приложенной

силой, следуя воздействию из внешней среды. Однако также видны и проблемы: отставание реакции робота примерно на 0.2 секунды и прерывистое перемещение звена. Как первую, так и вторую проблему предполагается решить, при помощи модификации алгоритма обработки изменения сигнала (пересмотр параметров формулы (1), частоты опроса датчиков силы, или же обработка с использованием нейросети).

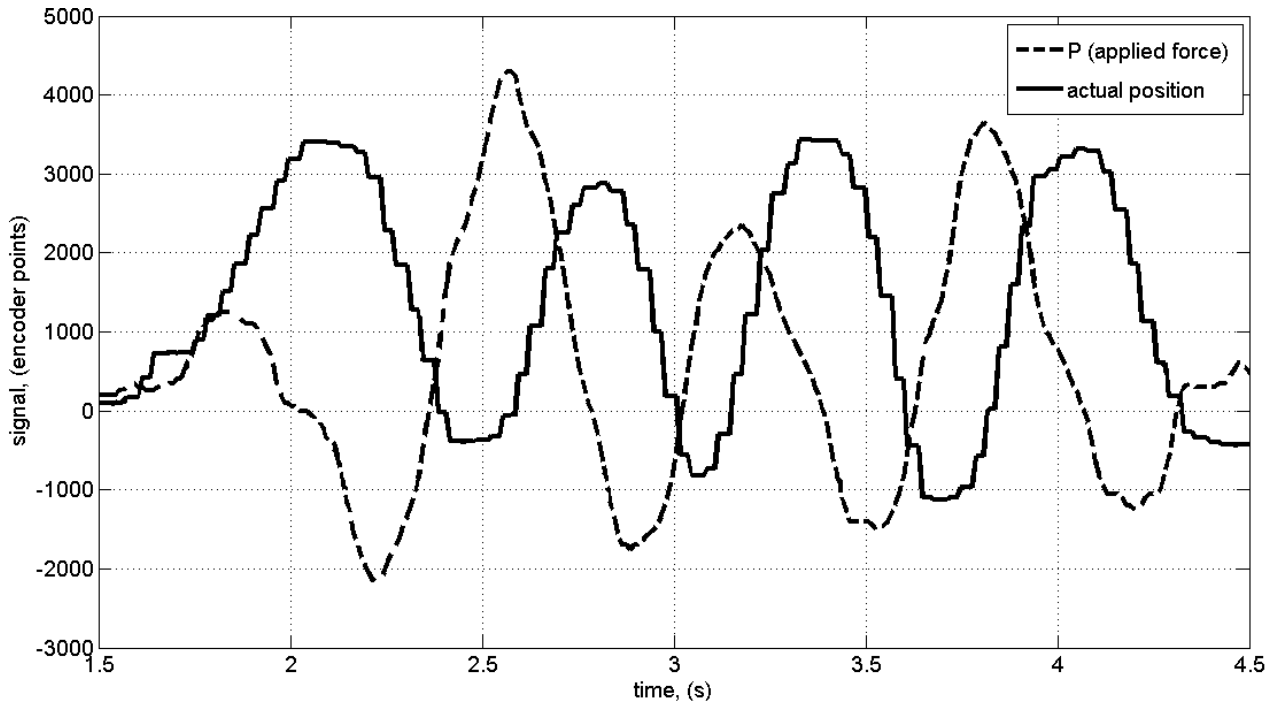


Рисунок 4 - График результатов эксперимента

Выводы

В ходе научной работы был создан программный интерфейс для управления роботами-манипуляторами Katana для операционных систем Linux и Windows. Программный интерфейс предоставляет средства для разработки управляющих программ, интерфейса пользователя, разнообразных приложений в учебных и профессиональных целях. Показана возможность управления роботом с помощью прямых физических манипуляций. Разработан алгоритм обработки сигналов и реализована модель поведения робота при взаимодействии с внешним миром. Направления дальнейшей работы: усовершенствование модели поведения робота с целью воплощения более естественных реакций на внешние воздействия, оснащение дополнительными анализаторами (таких как видеочамера).

Список литературы

1. Richard Gourdeau. A Robotics Object Oriented Package in C++. – Montreal: Polytechnique de Montréal, 2006. – 537 p.
2. А.А. Мельник, В.Н. Хоменко, П.С. Плис, П. Энафф, В.Ф. Борисенко. Кинематическая модель робота с шестью степенями свободы и возможностью учета зазора в суставах/Наукові праці Донецького національного технічного університету. - №10 (180), 2011. - с.113-120.
3. Vincent Padois, Commande des Systèmes Robotiques – NSR13. Modélisation. – Paris: UPMC: Université Pierre et Marie Curie (CNRS UMR 7222), 2007. – 37 p.
4. Fundamentals of Robotics. Linking Perception to Action. H. Bunke, P. S. P. Wang. – New Jersey: World Scientific, 2004. – 718 p.