

Math-Net.Ru

All Russian mathematical portal

A. A. Ardentov, A. V. Smirnov, Controlling a mobile robot along Euler's elasticae, *Program Systems: Theory and Applications*, 2017, Volume 8, Issue 4, 163–178

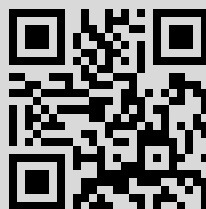
DOI: <https://doi.org/10.25209/2079-3316-2017-8-4-163-178>

Use of the all-Russian mathematical portal Math-Net.Ru implies that you have read and agreed to these terms of use
<http://www.mathnet.ru/eng/agreement>

Download details:

IP: 213.24.127.4

March 14, 2018, 12:13:02



А. А. Ардентов, А. В. Смирнов
**Управление мобильным роботом
вдоль эластик Эйлера**

Аннотация. В данной работе рассматривается задача управления колесным роботом на плоской поверхности без препятствий. В основе сборки робота лежит платформа Arduino. Исследуется оптимальный класс управлений, впервые возникший еще в классической задаче Эйлера об эластиках. В конце работы приведено несколько примеров перемещения робота вдоль эластик Эйлера, изучена проблема ухода траектории от элаستيку и возможные пути её решения.

Ключевые слова и фразы: Мобильный робот, кинематика, эластика Эйлера, Ардуино.

Введение

Развитие компьютерной техники в конце прошлого столетия подтолкнуло исследователей к созданию различных моделей управления мобильными роботами. Существенный вклад в развитие этого направления внес Ж.-П. Ломонд, вместе со своими коллегами в работе [1] он описывает общий подход планирования движения робота при наличии неголономных связей. Простейшей неголономной связью является гипотеза «прокручивания колеса без проскальзывания», которую следует принимать при моделировании движения колесных роботов. Она определяет дифференциальные связи, задающие кинематическую модель робота.

Примерно в то же время возникает современная геометрическая теория управления [2], которая позволяет сформулировать задачу оптимального управления на языке дифференциального и интегрального исчисления и решить её с помощью геометрических методов. Машина Дубинса [3] и машина Ридса–Шеппа [4] являются простейшими математическими моделями робота, которые определяются задачами быстройдействия. В обеих моделях присутствует ограничение на

Работа выполнена при финансовой поддержке гранта Российского научного фонда (грант №17-11-01387) в Институте программных систем им. А.К. Айламазяна РАН.

© А. А. Ардентов, А. В. Смирнов, 2017

© Институт программных систем имени А. К. Айламазяна РАН, 2017

© Программные системы: теория и приложения, 2017

DOI: 10.25209/2079-3316-2017-8-4-163-178

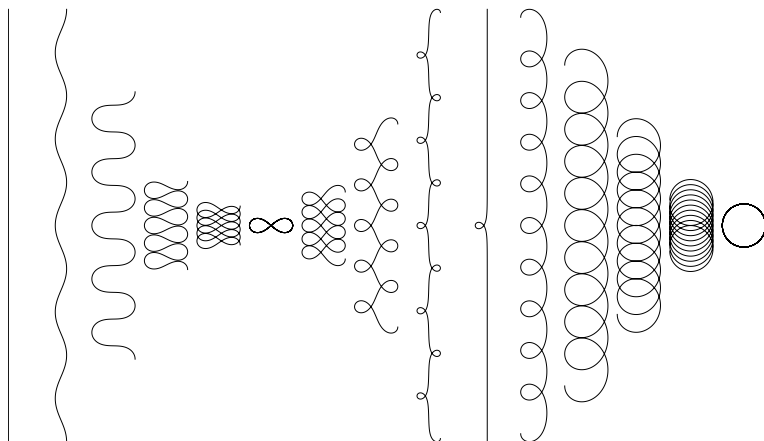


Рис. 1: Семейство эластик Эйлера: от прямой до окружности

угловую скорость, определяющее минимальный радиус окружности, по которой может двигаться робот, для построения кратчайшей траектории между произвольными состояниями машины используются сегменты прямых и дуги окружностей.

В данной работе исследуется модель управления роботом на основе эластик Эйлера, которые впервые были описаны Леонардом Эйлером в 1744 г. [5]. Это семейство оптимальных кривых, которые минимизируют интеграл квадрата кривизны кривой, оно включает в себя прямую и окружность, все семейство приведено на рис. 1. В работе [6] Дэвид Мамфорд мотивировал использование эластик в компьютерном зрении для восстановления скрытого контура. Оказывается, что эластика является наиболее вероятной траекторией броуновского движения из одного положения на плоскости в другое с заданными направлениями в них при условии, что длины рассматриваемых траекторий распределены экспоненциально. Предлагается рассмотреть математическую модель робота,двигающегося с постоянной линейной скоростью, для которого вдоль пути минимизируется интеграл от квадрата угловой скорости, что соответствует минимизации маневров робота. В такой модели движение происходит вдоль эластик Эйлера. Интерфейс для моделирования эластик Эйлера в программной среде Mathematica представлен в работе [7].

1. Математическая модель управления роботом

Движение робота описывается следующей дифференциальной системой:

$$(1) \quad \begin{cases} \dot{x} = u_t \cos \theta, \\ \dot{y} = u_t \sin \theta, \\ \dot{\theta} = u_a, \end{cases}$$

где x, y определяют координаты центра робота в плоскости движения, угол θ задает направление движения колёс; параметр u_t соответствует линейной скорости, а u_a — угловой.

Аналогичная система [8] возникает в задаче Эйлера об эластике, в которой параметр линейной скорости постоянен $u_t \equiv \text{const}$. Решением этой задачи являются кривые, локально минимизирующие функционал

$$(2) \quad \int u_a^2(t) dt \rightarrow \min,$$

параметр u_a соответствует кривизне искомой кривой, называемой эластикой Эйлера. В работе [9] приведена параметризация эластик Эйлера, а также формулы вычисления управления u_a для разных типов эластик. Для моделирования движения робота вдоль эластике Эйлера использовались формулы вычисления скоростей колес [1]:

$$(3) \quad u_l = u_t - \frac{u_a l}{2}, \quad u_r = u_t + \frac{u_a l}{2}.$$

где u_l, u_r суть скорости левого и правого колеса робота, а l есть расстояние между колесами.

2. Описание прототипа колёсного робота

Колёсный робот построен на основе платформы из оргстекла, имеющей двухколёсный привод (2WD). Преимущество данной платформы — малое энергопотребление при маневрах, что позволяет значительно увеличить ее грузоподъемность, а ее большую рабочую поверхность использовать для различных целей, например, для установки еще большего количества датчиков сенсоров и др.

В качестве бортового вычислителя используется плата Arduino Mega 2560 [10]. Arduino — марка аппаратно-программных средств для создания простых систем автоматики и робототехники [11, 12]. Программная часть состоит из бесплатной программной оболочки (IDE)

для написания программ, их компиляции и программирования аппаратуры. Аппаратная часть представляет собой набор смонтированных печатных плат, продающихся как официальным производителем, так и сторонними производителями. Arduino может использоваться как для создания автономных объектов автоматики, так и подключаться к программному обеспечению на компьютере через стандартные проводные и беспроводные интерфейсы. Arduino и Arduino-совместимые платы спроектированы таким образом, чтобы их можно было при необходимости расширять, добавляя в устройство новые компоненты. Эти платы расширений подключаются к Arduino посредством установленных на них штыревых разъёмов. Существует ряд плат с унифицированным конструктивом, допускающим конструктивно жесткое соединение процессорной платы и плат расширения в стопку через штыревые линейки [13].

Микроконтроллеры для Arduino отличаются наличием предварительно прошитого в них загрузчика (bootloader). С помощью этого загрузчика пользователь загружает свою программу в микроконтроллер без использования традиционных отдельных аппаратных программаторов. Загрузчик соединяется с компьютером через интерфейс USB (если он есть на плате) или с помощью отдельного переходника UART-USB. В линейке устройств Arduino в основном применяются микроконтроллеры Atmel AVR ATmega328, ATmega168, ATmega2560, ATmega32U4, ATTiny85 с частотой тактирования 16 или 8 МГц. В старых изделиях применялись ATmega8, ATmega1280 и другие. Программирование ведется целиком через собственную программную оболочку (IDE), бесплатно доступную на сайте Arduino. В этой оболочке имеется текстовый редактор, менеджер проектов, препроцессор, компилятор и инструменты для загрузки программы в микроконтроллер. Оболочка написана на Java на основе проекта Processing, работает под Windows, MacOSX и Linux. Закачка программы в микроконтроллер Arduino происходит через предварительно запрограммированный специальный загрузчик. Загрузчик создан на основе AVR109 [14]. Загрузчик может работать через интерфейсы RS-232, USB или Ethernet в зависимости от состава периферии конкретной процессорной платы.

На платформе установлены два мотор-редуктора постоянного тока с рабочим напряжением 9 вольт и максимальной частотой вращения 150 оборотов в минуту. На моторах смонтированы датчики Холла для мониторинга и контроля скорости вращения. Оба мотора подключены к специальной плате — драйверу L298N. Данная

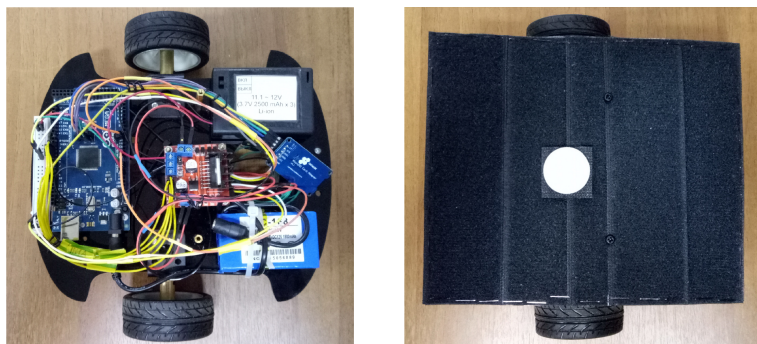


Рис. 2: Прототип колёсного робота в собранном состоянии и «панцирь» с маркером, закреплённый на роботе

плата позволяет управлять скоростью вращения, используя принцип широтно-импульсной модуляции (ШИМ) и подключена к плате Arduino. Также, к Arduino подключён специальный адаптер для чтения данных с SD карты. Для питания робота используются два аккумулятора на 12 вольт, один питает бортовой вычислитель, другой подсистему моторов.

Робот имеет три колеса. Два основных имеют диаметр 65 мм и установлены на моторах. Третье, шариковое колесо, установлено в задней части робота. Собранный прототип колёсного робота представлен слева на рис. 2.

2.1. Управление роботом

Управление движением робота осуществляется с помощью специальной микропрограммы — скетча, загруженного на контроллер. Изменение скоростей вращения валов двигателей осуществляется с помощью широтно-импульсной модуляции (ШИМ) [15]. Микропрограмма считывает данные о траектории движения из специального файла, записанного на SD карту. Формат данных следующий: время, ШИМ для левого двигателя, ШИМ для правого двигателя. В самом простом случае, файл может содержать следующую информацию:

0,70,70,3,0,0.

Она означает движение по прямой в течении трёх секунд с ШИМ равным 70 на обоих двигателях. Таким образом можно рассчитать и

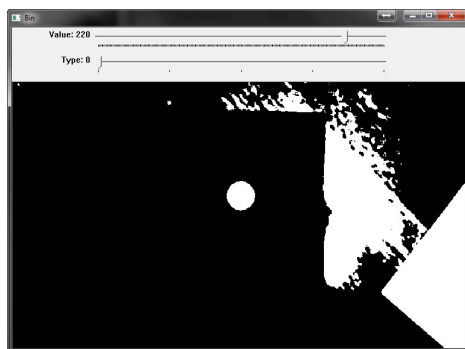


Рис. 3: Бинаризованное изображение

задать более сложные траектории, например, движение по «восьмёрке».

2.2. Отслеживание движения робота

Отслеживание движения робота предполагает сравнение фактической траектории движения робота с рассчитанной и записанной в файл. Для фиксирования фактической траектории был разработан алгоритм детектирования специального маркера, установленного на борту робота. В качестве маркера был выбран белый круг. Данный маркер закрепляется на чёрном «панцире» робота. «Панцирь» позволяет закрепить несколько подобных маркеров, для более точного определения положения робота. Сам «панцирь» закрепляется на верхней части робота. «Панцирь» с маркером представлен справа на рис. 2.

Алгоритм детектирования маркера достаточно прост и состоит из двух основных этапов:

- (1) пороговая бинаризация изображения,
- (2) контурный анализ бинарного изображения.

Бинаризация изображений [16–18], то есть перевод полноцветного или в градациях серого изображения в монохромное, где присутствуют только два типа пикселей (чёрные и белые) имеет большое значение при распознавании образов. Пороговые методы бинаризации работают со всем изображением, находя какую-то характеристику (порог), позволяющую разделить все изображение на чёрное и белое. При бинаризации изображения, также можно использовать различные

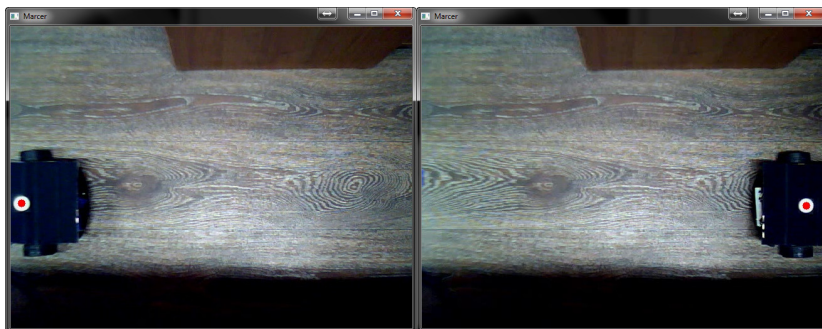


Рис. 4: Примеры детектирования маркера

фильтры, сглаживающие изображение, что помогает избавиться от шумов. Результат пороговой бинаризации показан на рис. 3, пороговое значение подобрано экспериментально.

Далее выполняется контурный анализ изображения. Контурный анализ [19] — это один из важных и очень полезных методов описания, хранения, распознавания, сравнения и поиска графических образов (объектов). Контур — это граница объекта, которая отделяет его от фона (других объектов). Переход к рассмотрению только контуров объектов позволяет уйти от пространства изображения к пространству контуров, что существенно снижает сложность алгоритмов и вычислений. При проведении контурного анализа:

- полагается, что контур содержит достаточную информацию о форме объекта;
- внутренние точки объекта во внимание не принимаются.

Бинарное изображение явным образом задаёт границы объекта. Совокупность пикселей, составляющих границу объекта, и есть контур объекта. Библиотека компьютерного зрения OpenCV [20] предоставляет функции для определения таких полезных свойств найденных контуров, как площадь. Значение площади контура позволяет определить круг минимальной площади, содержащий данный контур. Далее находится отношение площади контура к площади круга. Соответственно, чем ближе их значения, тем больше контур похож на круг, то есть на искомый маркер. В результате действия алгоритма центр маркера помечается красной точкой, см. рис. 4.

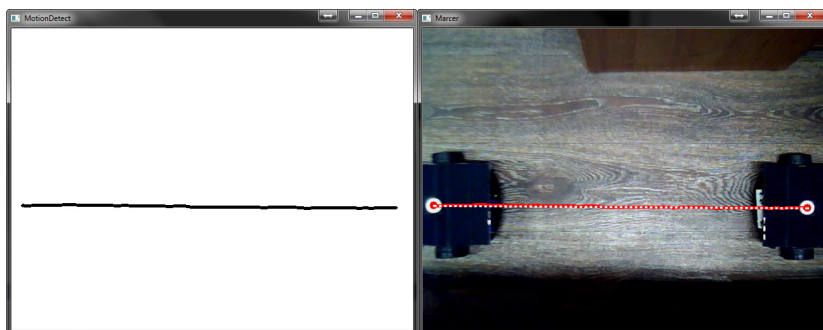


Рис. 5: Траектория движения робота вдоль прямой, совмещенное изображение

Для тестирования алгоритма детектирования маркера, в робот была заложена одна из самых простых задач — движение по прямой. Чтобы отследить фактическую траекторию, была использована обыкновенная Web-камера с разрешением съёмки 640 на 480 пикселей. Камера была закреплена перпендикулярно относительно плоскости движения робота. Слева на рис. 5 изображена траектория движения вдоль прямой, справа приведено совмещающее в себе начальное и конечное положение, красным цветом отмечена траектория движения робота, белый пунктир соответствует прямой линии между начальным и конечным положением. Ниже в разделе 3 для краткости приведены лишь совмещенные изображения. Из рис. 5 видно, что робот двигается по прямой линии достаточно точно, хотя и немного смещается от заданного курса. Это может быть следствием воздействия возникшей силы трения с покрытием, по которому двигается робот и которое не учитывается в микропрограмме. Также на движение робота могут влиять разные значения оборотов моторов, которые возникают из-за их конструктивных особенностей.

В следующем разделе приведены результаты экспериментов — движение вдоль различных эластик Эйлера.

3. Примеры движений робота вдоль эластик Эйлера

Простейшими формами эластик являются отрезки прямых и дуги окружностей, движение по которым осуществляется при постоянных скоростях. В случае $u_l = u_r \neq 0$ робот двигается вдоль прямой. Если

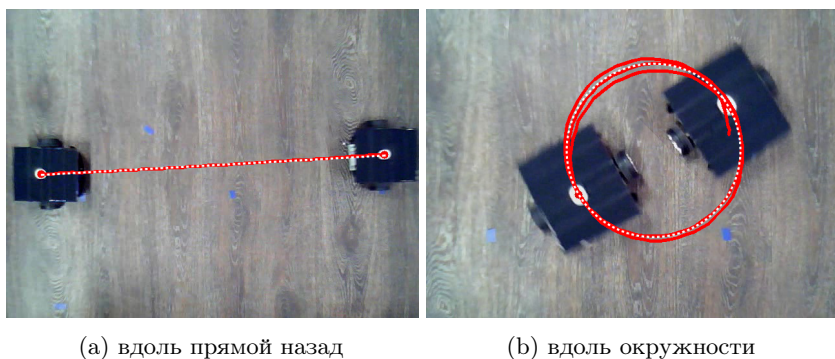


Рис. 6: Движение колёсного робота с постоянными скоростями

скорость правого колеса отличается от скорости левого, то движение происходит по дуге окружности.

На рис. 6а приведено совмещенное изображение движения робота назад вдоль прямой. Из рисунка видно, что в начале робот поворачивается на небольшой угол, а потом двигается вдоль прямой достаточно точно. Это обусловлено отличием двигателей для левого и правого колеса, ярко выраженным во время старта. Отметим, что эта тенденция сохраняется и для всех последующих тестов.

На рис. 6б изображены два совмещенных в одном кадре промежуточных состояния робота при движении вдоль окружности. Красным цветом изображена фактическая траектория движения, а белым пунктиром — окружность, проходящая через выбранные положения робота, определяющие её диаметр. Из рисунка видно, что в середине движения робот движется достаточно близко к окружности.

Скорость правого колеса для движения вдоль «восьмёрки» вычислялась по следующей формуле [9]:

$$(4) \quad u_r = u_t - k_0 l \operatorname{cn} (K(k_0) + t, k_0),$$

где константа $k_0 \approx 0.909$, функция K задает полный эллиптический интеграл Лежандра I рода, а функция cn соответствует эллиптическому косинусу (напомним, что управление u_a для эластик определяется с помощью эллиптических функций Якоби и эллиптических интегралов [21]). Аналогичным образом определяется скорость левого колеса. Так как скорость робота определяется целым значением ШИМ, то соответствующие функции u_r , u_l были дискретизированы. На рис. 7а

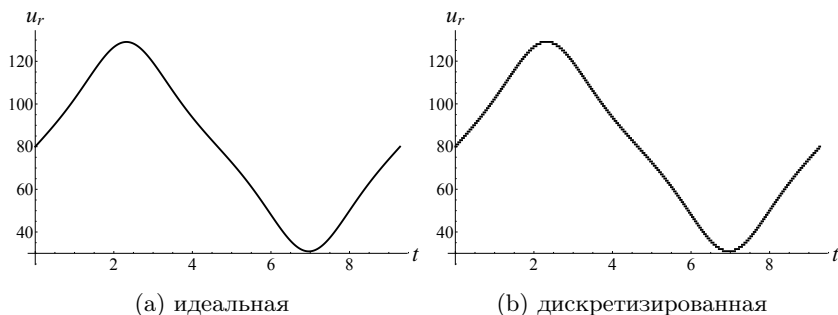


Рис. 7: Скорость колеса при движении вдоль «восьмёрки»

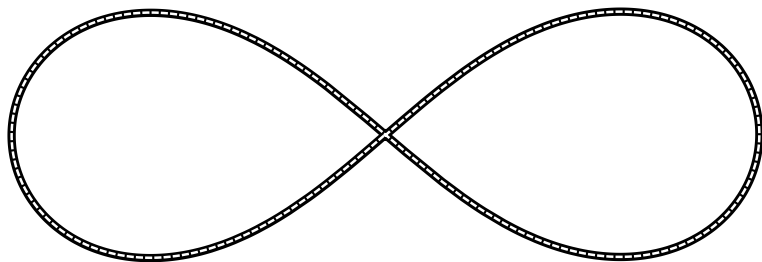


Рис. 8: Сравнение «восьмёрок» для непрерывного (черная) и дискретного управления (белая пунктирная)

приведена функция u_r , а на рис. 7b её целочисленное приближение $\text{Round}(u_r)$.

Численное интегрирование системы (1) показывает, что функции для идеальной и дискретизированной скоростей колес, приведенные на рис. 7, определяют неотличимые на глаз эластики в форме «восьмёрки», см. сравнение на рис. 8.

На рис. 9 приведены четыре примера движения вдоль инфлекссионной эластики в форме «восьмёрки». В каждом примере было совмещено 4 кадра из движения, в том числе начальное положение робота, траектория движения и соответствующая идеальная инфлекссионная эластика. Из результатов тестирования, приведенных на рис. 9, видно, что вторая половина траекторий совпадает с «восьмёркой» в каждом примере.

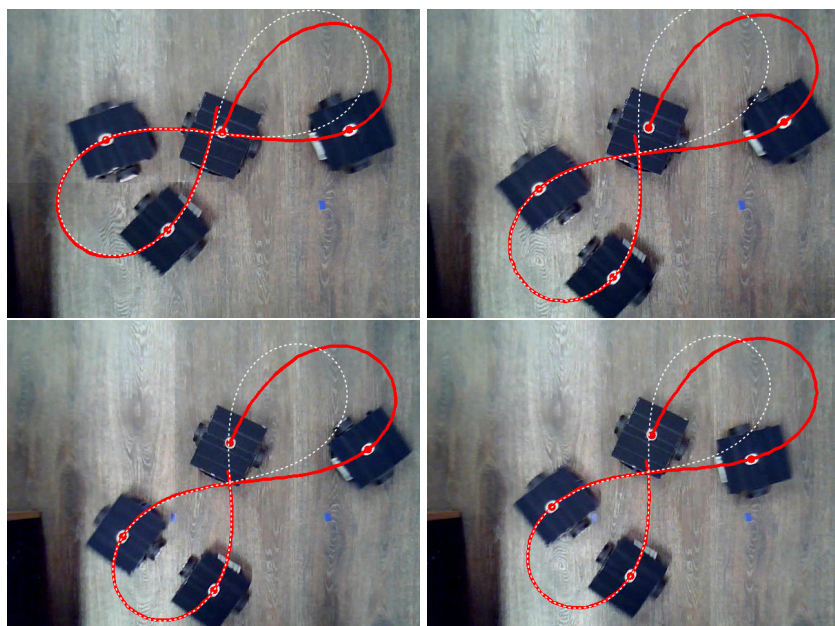


Рис. 9: Примеры движения вдоль «восьмёрки»



Рис. 10: Пример движения вдоль неинфлексионной элаستي

Наконец, заснято движение робота вдоль неинфлексионной эластике, см. рис. 10. Локально траектория совпадает с исходной эластикой, однако глобально траектория заметно уходит от эталона.

4. Заключение

В данной работе рассмотрена задача управления колесным роботом вдоль эластик Эйлера. Разработана программа на языке Wiring, которая принимает на вход файл с данными для скоростей колес робота. Программа отлажена на простейших формах эластик — прямых и окружностях, а также на более сложных формах инфлекссионных и неинфлексионных эластик общего вида.

В ходе тестирования выявлены следующие проблемы:

- так как поверхность движения неровная, то возникает эффект проскальзывания колес;
- двигатели правого и левого колеса не являются абсолютно идентичными, требуется точная калибровка, особенно при старте и остановке двигателей;
- не учитывается сила трения колес с поверхностью.

Для их устранения требуется более сложный подход, использующий принцип обратной связи и учитывающий вероятностный характер динамического поведения [22]. С помощью датчиков робот должен определять реальную скорость каждого колеса, соотносить её с требуемой и при необходимости увеличивать или уменьшать ШИМ.

На основе модели, представленной в данной работе, планируется разработать алгоритм оптимальной парковки робота на плоскости, в том числе с препятствиями. Схожий алгоритм можно будет использовать для управления колесным роботом с прицепом. Известно [23], что конкатенации дуг эластик Эйлера являются кратчайшими в задаче быстрогодействия для колесного робота с прицепом при наличии ограничения на управления.

Список литературы

- [1] J.-P. Laumond, *Nonholonomic Motion Planning for Mobile Robots*, Tutorial Notes, LAAS-CNRS, Toulouse, 1998. ↑ [163,165](#)
- [2] V. Jurdjevic. *Geometric Control Theory*, Cambridge University Press, 1997, 492 p. ↑ [163](#)

- [3] L. E. Dubins. “On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents”, *American Journal of Mathematics*, **79**:3 (1957), pp. 497–516. [↑] [163](#)
- [4] J. A. Reeds, L. A. Shepp. “Optimal paths for a car that goes both forwards and backwards”, *Pacific J. Math.*, **145**:2 (1990), pp. 367–393. [↑] [163](#)
- [5] L. Euler. *Methodus inveniendi lineas curvas maximi minimive proprietate gaudentes, sive solutio problematis isoperimetrici latissimo sensu accepti*, Bousquet, Lausanne, 1744. [↑] [164](#)
- [6] D. Mumford. “Elastica and Computer Vision”, *Algebraic Geometry and its Applications*, ed. C. L. Bajaj, Springer, New York, 1994, pp. 491–506. [↑] [164](#)
- [7] А. А. Ардентов. «Интерфейс для моделирования эластик Эйлера в программной среде Mathematica», *Программные системы: теория и приложения*, **3**:1 (2012), с. 31–50, URL: http://psta.psir.ru/read/psta2012_1_31-50.pdf [↑] [164](#)
- [8] A. P. Mashtakov, A. A. Ardentov, Yu. L. Sachkov. “Relation Between Euler’s Elasticae and Sub-Riemannian Geodesics on $SE(2)$ ”, *Regular and Chaotic Dynamics*, **21**:7–8 (2016), pp. 832–839. [↑] [165](#)
- [9] Yu. L. Sachkov. “Maxwell strata in Euler’s elastic problem”, *Journal of Dynamical and Control Systems*, **14**:2 (2008), pp. 169–234. [↑] [165](#), [↑] [171](#)
- [10] *Arduino Mega 2560*, <http://arduino.ru/Hardware/ArduinoBoard-Mega2560>. [↑] [165](#)
- [11] A. Elfasakhany, E. Yanez, K. Baylon, R. Salgado. “Design and Development of a Competitive Low-Cost Robot Arm with Four Degrees of Freedom”, *Modern Mechanical Engineering*, **1**:2 (2011), pp. 47–55. [↑] [165](#)
- [12] R. Eide, P. Egelid, A. Stamsø, H. Karimi. “LQG Control Design for Balancing an Inverted Pendulum Mobile Robot”, *Intelligent Control and Automation*, **2**:2 (2011), pp. 160–166. [↑] [165](#)
- [13] В. А. Петин. *Проекты с использованием контроллера Arduino*, БХВ-Петербург, СПб., 2014, 400 с. [↑] [166](#)
- [14] *AVR109: Self Programming*, 2004, URL: <http://www.atmel.com/images/doc1644.pdf> [↑] [166](#)
- [15] *Преобразователи частоты*, Элементная база и схемотехника устройств силовой электроники, URL: <http://leg.co.ua/knigi/raznoe/elementnaya-baza-i-shemotehnika-ustroystv-silovoy-elektroniki-4.html> [↑] [167](#)
- [16] *Conversion of a Color Image to a Binary Image*, 2005, URL: <http://www.CoderSource.net> [↑] [168](#)
- [17] У. Прэтт. *Цифровая обработка изображений*. Т. 1,2, Наука, М., 2000, 1024 с. [↑] [168](#)
- [18] А. Фёдоров. *Бинаризация черно-белых изображений: состояние и перспективы развития*, URL: <http://it-claim.ru/Library/Books/ITS/wwwbook/ist4b/its4/fyodorov.htm> [↑] [168](#)

- [19] Я. А. Фурман (ред.). *Введение в контурный анализ. Приложения к обработке изображений и сигналов*, Физматлит, М., 2003, 592 с. ↑¹⁶⁹
- [20] G. Bradski, A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*, O'Reilly Media, 2008, 556 p. ↑¹⁶⁹
- [21] Н. И. Ахиезер. *Элементы теории эллиптических функций*, Наука, М., 1970, 304 с. ↑¹⁷¹
- [22] А. В. Борисов, И. С. Мамаев. «Динамика саней Чаплыгина», *Прикладная математика и механика*, **73**:2 (2009), с. 219–225. ↑¹⁷⁴
- [23] Н. Chitsaz. “On time-optimal trajectories for a car-like robot with one trailer”, *SIAM Conf. on Control and its Applications*, 2013, pp. 114–120. ↑¹⁷⁴

Пример ссылки на эту публикацию:

А. А. Ардентов, А. В. Смирнов. «Управление мобильным роботом вдоль эластик Эйлера», *Программные системы: теория и приложения*, 2017, **8**:4(35), с. 163–178. URL: http://psta.psiras.ru/read/psta2017_4_163-178.pdf

Об авторах:



Андрей Андреевич Ардентов

Старший научный сотрудник Исследовательского Центра Процессов Управления Института Программных Систем имени А. К. Айламазяна РАН

e-mail:

aaa@pereslavl.ru



Александр Владимирович Смирнов

Инженер-исследователь Лаборатории методов обработки и анализа изображений, Институт Программных Систем имени А. К. Айламазяна РАН

e-mail:

asmirnov_1991@mail.ru

Andrei Ardentov, Alexander Smirnov. *Controlling a mobile robot along Euler's elasticae*.

ABSTRACT. In this paper, we consider the problem of controlling a wheeled robot on a flat surface without obstacles. Arduino platform is a basis of the robot. We investigate the class of optimal controls that first appeared in the classical Euler's elastic problem. Finally, we present several examples of movement of the robot along Euler's elasticae and study the problem of trajectory avoidance from elasticae and possible ways to solve it. (*In Russian*).

Key words and phrases: mobile robot, kinematics, Euler's elastica, Arduino.

References

- [1] J.-P. Laumond, *Nonholonomic Motion Planning for Mobile Robots*, Tutorial Notes, LAAS-CNRS, Toulouse, 1998.
- [2] V. Jurdjevic. *Geometric Control Theory*, Cambridge University Press, 1997, 492 p.
- [3] L. E. Dubins. "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents", *American Journal of Mathematics*, **79**:3 (1957), pp. 497–516.
- [4] J. A. Reeds, L. A. Shepp. "Optimal paths for a car that goes both forwards and backwards", *Pacific J. Math.*, **145**:2 (1990), pp. 367–393.
- [5] L. Euler. *Methodus inveniendi lineas curvas maximi minimive proprietate gaudentes, sive solutio problematis isoperimetrici latissimo sensu accepti*, Bousquet, Lausanne, 1744.
- [6] D. Mumford. "Elastica and Computer Vision", *Algebraic Geometry and its Applications*, ed. C. L. Bajaj, Springer, New York, 1994, pp. 491–506.
- [7] A. A. Ardentov. "Euler's elasticae interface in Mathematica", *Program Systems: Theory and Applications*, **3**:1 (2012), pp. 31–50 (in Russian), URL: http://psta.psir.ru/read/psta2012_1_31-50.pdf
- [8] A. P. Mashtakov, A. A. Ardentov, Yu. L. Sachkov. "Relation Between Euler's Elasticae and Sub-Riemannian Geodesics on $SE(2)$ ", *Regular and Chaotic Dynamics*, **21**:7–8 (2016), pp. 832–839.
- [9] Yu. L. Sachkov. "Maxwell strata in Euler's elastic problem", *Journal of Dynamical and Control Systems*, **14**:2 (2008), pp. 169–234.
- [10] *Arduino Mega 2560*, <http://arduino.ru/Hardware/ArduinoBoardMega2560>.
- [11] A. Elfassakhany, E. Yanez, K. Baylon, R. Salgado. "Design and Development of a Competitive Low-Cost Robot Arm with Four Degrees of Freedom", *Modern Mechanical Engineering*, **1**:2 (2011), pp. 47–55.
- [12] R. Eide, P. Egelid, A. Stamsø, H. Karimi. "LQG Control Design for Balancing an Inverted Pendulum Mobile Robot", *Intelligent Control and Automation*, **2**:2 (2011), pp. 160–166.
- [13] V. A. Petin. *Projects using Arduino controller*, BKhV-Peterburg, SPb., 2014 (in Russian), 400 p.

- [14] *AVR109: Self Programming*, 2004, URL: <http://www.atmel.com/images/doc1644.pdf>
- [15] *Frequency converters*, Element base and circuitry of devices of power electronics (in Russian), URL: <http://leg.co.ua/knigi/raznoe/elementnaya-baza-i-shemotehnika-ustroystv-silovoy-elektroniki-4.html>
- [16] *Conversion of a Color Image to a Binary Image*, 2005, URL: <http://www.CoderSource.net>
- [17] W. K. Pratt. *Digital image processing*, 2nd ed., John Wiley & Sons, Inc., New York, 1991.
- [18] A. Fedorov. *Binarization of black and white images: state and development prospects* (in Russian), URL: <http://it-claim.ru/Library/Books/ITS/wwwbook/ist4b/its4/fyodorov.htm>
- [19] Ya. A. Furman (red.). *Introduction to the contour analysis. Applications for image and signal processing*, Fizmatlit, M., 2003 (in Russian), 592 p.
- [20] G. Bradski, A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*, O'Reilly Media, 2008, 556 p.
- [21] N. I. Akhiezer, *Elements of the Theory of Elliptic Functions*, Translations of Mathematical Monographs, American Mathematical Society, Providence, 1990, 237 p.
- [22] A. V. Borisov, I. S. Mamayev. “The dynamics of a Chaplygin sleigh”, *Journal of Applied Mathematics and Mechanics*, **73**:2 (2009), pp. 156–161.
- [23] H. Chitsaz. “On time-optimal trajectories for a car-like robot with one trailer”, *SIAM Conf. on Control and its Applications*, 2013, pp. 114–120.

Sample citation of this publication:

Andrei Ardentov, Alexander Smirnov. “Controlling a mobile robot along Euler’s elasticae”, *Program systems: Theory and applications*, 2017, **8**:4(35), pp. 163–178. (In Russian).

URL: http://psta.psiras.ru/read/psta2017_4_163-178.pdf