



УДК 004.942, 519.876.5

<https://doi.org/10.37661/1816-0301-2021-18-4-53-68>

Оригинальная статья
Original Paper

Оптимизация коэффициентов ПИД-регулятора системы управления движением мобильного робота по цветоконтрастной линии на основе генетического алгоритма

Т. Ю. Ким[✉], Г. А. Прокопович

Объединенный институт проблем информатики
Национальной академии наук Беларуси,
ул. Сурганова, 6, Минск, 220012, Беларусь
[✉]E-mail: tatyana_kim92@mail.ru

Аннотация

Цели. Разработать систему управления движением мобильного робота по цветоконтрастной линии, а также найти значения коэффициентов пропорционально-интегрально-дифференцирующего (ПИД-) регулятора, позволяющего роботу двигаться вдоль линии с заданной скоростью.

Методы. Для настройки значений коэффициентов ПИД-регулятора используются методы перебора, автоматической настройки и генетический алгоритм.

Результаты. Разработан программный комплекс настройки ПИД-регулятора учебного мобильного робота RoboCoke, предназначенного для движения по замкнутой цветоконтрастной линии с заданной скоростью. Программный комплекс состоит из имитационной модели указанного робота в среде Simulink, четырех виртуальных трасс-полигонов и специализированного решателя на базе разработанного генетического алгоритма. С помощью предложенной функции пригодности реализована система управления мобильным роботом, удовлетворяющая поставленным условиям. На основе проведенных модельных экспериментов получены искомые значения параметров ПИД-регулятора.

Заключение. Проведено сравнение эффективности различных методов настройки ПИД-регулятора. Разработанный программный комплекс предназначен для решения практической задачи движения мобильного робота по цветоконтрастной линии со скоростью 1 м/с. Полученные результаты могут быть применены для исследования методов эволюционной настройки систем стабилизации транспортных роботов, обеспечивающих их движение без перерегулирования.

Ключевые слова: система стабилизации движения, генетический алгоритм, ПИД-регулятор, имитационная модель, имитационное моделирование, обратная связь, сенсоры

Благодарности. Работа выполнялась в рамках проекта БРФФИ Ф20Р-324 и задания Т31 ГПНИ «Цифровые и космические технологии, безопасность человека, общества и государства» (2021–2025).

Для цитирования. Ким, Т. Ю. Оптимизация коэффициентов ПИД-регулятора системы управления движением мобильного робота по цветоконтрастной линии на основе генетического алгоритма / Т. Ю. Ким, Г. А. Прокопович // Информатика. – 2021. – Т. 18, № 4. – С. 53–68.

<https://doi.org/10.37661/1816-0301-2021-18-4-53-68>

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Поступила в редакцию | Received 04.10.2021
Подписана в печать | Accepted 10.11.2021
Опубликована | Published 29.12.2021

Optimization of the PID coefficients for the line-follower mobile robot controller employing genetic algorithm

Tatyana Yu. Kim[✉], Ryhor A. Prakapovich

*The United Institute of Informatics Problems
of the National Academy of Sciences of Belarus,
st. Surganova, 6, Minsk, 220012, Belarus
✉E-mail: tatyana_kim92@mail.ru*

Abstract

Objectives. To develop a control system for the movement of a mobile robot along a color-contrast line, as well as to find the values of the coefficients of a proportional-integral-differentiating (PID) controller that allows the robot to move along the line at a given speed.

Methods. To adjust the values of the coefficients of the PID controller, methods of enumeration, automatic tuning and a genetic algorithm are used.

Results. A software package for tuning the PID controller of the educational mobile robot RoboCake, designed to move along a closed color-contrast line at a given speed, has been developed. The software package consists of a simulation model of the specified robot in the Simulink environment, several virtual traces-polygons and a specialized solver based on the developed genetic algorithm. With the help of the proposed fitness function, a mobile robot control system that satisfies the stated conditions is implemented. Based on the conducted model experiments, the desired values of the parameters of the PID controller are obtained.

Conclusion. A comparison of the effectiveness of various methods of tuning the PID controller is carried out. The developed software package is designed to solve the practical problem of moving a mobile robot along a color-contrast line at a speed of 1 m/s. The results obtained can be used to study methods of evolutionary tuning of stabilization systems for transport robots, ensuring their movement without overshoot.

Keywords: motion stabilization system, genetic algorithm, PID controller, simulation model, simulation, feedback, sensors

Acknowledgements. The work was carried out within the framework of the BRFFR project F20R-324 and the task T31 of the State Scientific Research Program "Digital and Space Technologies, Human, Society and State Security" (2021–2025).

For citation. Kim T. Yu., Prakapovich R. A. *Optimization of the PID coefficients for the line-follower mobile robot controller employing genetic algorithm*. Informatika [Informatics], 2021, vol. 18, no. 4, pp. 53–68 (In Russ.). <https://doi.org/10.37661/1816-0301-2021-18-4-53-68>

Conflict of interest. The authors declare of no conflict of interest.

Введение. Наиболее исследованы и часто применяются в промышленности ПИД-регуляторы, которые занимают до 90 % всего рынка [1, 2]. Как правило, точность и быстродействие ПИД-регуляторов являются приемлемыми для большинства современных задач по соотношению «цена – качество» и обеспечивают хорошую скорость решений для многих прикладных задач. Для настройки ПИД-регулятора требуется наличие аналитического описания объекта управления и его имитационной модели либо обеспечение возможности безопасного пробного запуска и тестовой эксплуатации самого объекта управления в различных режимах. Остальные 10 % рынка заняты узкоспециализированными регуляторами, которые требуют применения более сложных методов теории автоматического управления.

Для относительно несложных объектов управления ПИД-регуляторы могут подбираться либо вручную методом проб и ошибок (при условии, что это не приводит к аварийным ситуациям), либо с применением классических методов автоматической настройки ПИД-регуляторов. К классическим относятся, например, непараметрические методы настройки, основанные на принципе непрерывного цикла Циглера – Николса [3], и метод минимизации интеграла абсолютной погрешности [4]. Однако из-за специфики физических законов функционирования не-

которых объектов управления названные методы не всегда соответствуют требованиям вычислительной эффективности, скорости работы и качеству их настройки. Подобные исследования рассмотрены в работе [5], где были оптимизированы коэффициенты ПИД-регулятора для мобильного робота, следующего по линии.

Существует несколько программных продуктов для настройки ПИД-регуляторов. Они могут быть как специализированными, так и универсальными. К первым относятся, например, PID-expert [6] и ExperTune (URL: <https://lapshinvr.ru/articals/expertune.html>). Ко второму типу можно отнести библиотеку Simulink Control Design (URL: https://www.mathworks.com/help/pdf_doc/control/control_gs.pdf), которая является одним из многочисленных пакетов прикладных программ популярного математического обеспечения MATLAB. Библиотека Simulink Control Design способна автоматически настраивать ПИД-регуляторы одноконтурных и многоконтурных систем управления, что избавляет от необходимости ручного подбора параметров и многократного повторения реальных практических экспериментов. Однако для этого требуется наличие аналитических формул, которые в той или иной степени описывают физические процессы функционирования объектов управления. Если же по каким-либо причинам таких формул нет или объекты управления являются достаточно сложными, то существуют подходы, использующие имитационное моделирование, например, с помощью библиотеки Simscape Multibody. К сожалению, для указанных программных средств важным критерием является то, что исследуемая модель должна быть линейной. В этом случае автоматическая настройка реализуется с помощью метода интерактивной оценки параметров имитационной модели на основе данных отклика (Interactively Estimate Plant Parameters from Response Data; URL: <https://www.mathworks.com/help/control/getstart/interactively-estimate-plant-parameters-from-response-data.html>) приложения PID-Tuner. Существуют работы, в которых для подбора коэффициентов ПИД-регулятора дополнительно применяются методы эволюционного моделирования, в частности генетические алгоритмы [7–11].

В статье описывается опыт авторов по автоматической настройке коэффициентов ПИД-регулятора системы управления движением мобильного робота по цветоконтрастной линии с помощью его имитационной модели и генетического алгоритма. Для этого были разработаны оптический сенсор для обратной связи системы управления и функция пригодности, обеспечивающая выполнение условия минимизации расстояния между сенсором и краем линии.

Конструкция тестового мобильного робота RoboCake. В качестве объекта исследования был выбран мобильный робот RoboCake Beginner (рис. 1, *a*), задача которого заключалась в минимизации времени прохождения по нелинейной цветоконтрастной линии. Данная задача является типовой для мобильной робототехники и формулируется как движение по заранее известной траектории, которая задана в виде нанесенных на поверхность движения цветоконтрастных или магнитных полос либо в виде набора связанных пространственных точек на цифровых картах помещений.

Мобильный робот RoboCake Beginner имеет двухколесную платформу с дифференциальным приводом, оснащенную инкрементными энкодерами и двумя шарнирными опорами для устойчивости. Под корпусом робота установлены три датчика света для обнаружения цветоконтрастной линии. На рис. 1 эти датчики отмечены красным цветом. Зона расположения датчиков представляет собой равнобедренный треугольник с основанием 5,5 см и высотой 6 см, вершина которого расположена на расстоянии 2,5 см от центра платформы в сторону движения. Центр робота находится между двумя колесами, которые равноудалены от него на 4,25 см.

Рассмотрим актуальную задачу, являющуюся типичной при использовании транспортных тележек на складах и производствах. Требуется разработать систему управления движением мобильного робота,двигающегося по цветоконтрастной линии со скоростью 1 м/с. При этом цветоконтрастная линия должна находиться между двумя передними датчиками робота. Для обеспечения нормальной работы алгоритма движения данного робота ширина линии должна быть равной 2,4 см (рис. 2).

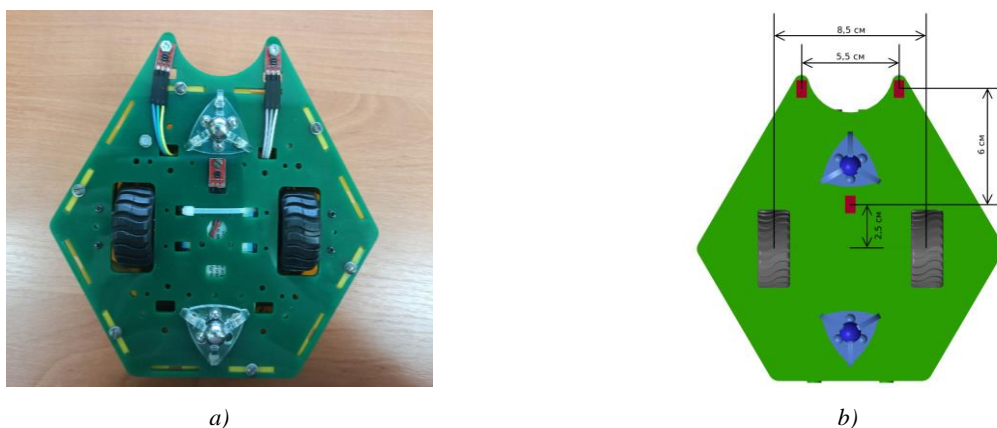


Рис. 1. Расположение датчиков света на нижней поверхности робота RoboCake Beginner (a) и его имитационной модели (b)

Fig. 1. Location of light sensors on the bottom of the robot RoboCake Beginner (a) and its simulation model (b)

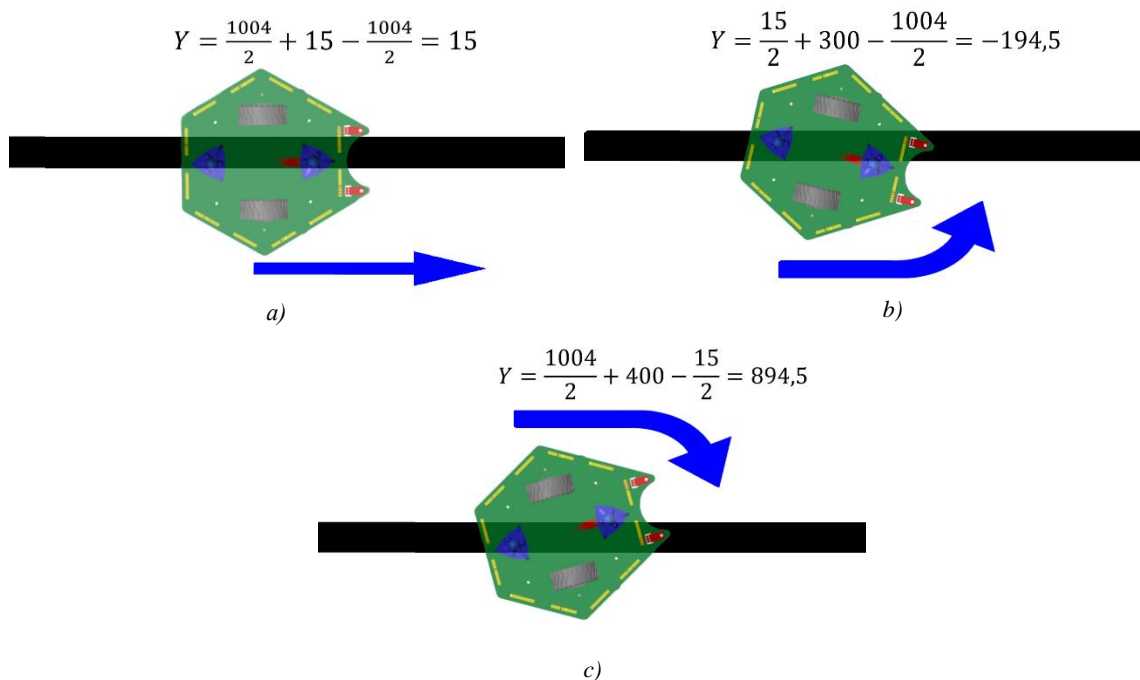


Рис. 2. Процесс корректировки положения мобильного робота относительно края линии в процессе его движения: a) вперед без изменений; b) вперед со смещением против часовой стрелки; c) вперед со смещением по часовой стрелке

Fig. 2. The process of correcting the position of the mobile robot relative to the edge of the line in the process of movement: a) move forward unchanged; b) move forward offset counterclockwise; c) forward offset clockwise

Чтобы определить положение робота RoboCake Beginner над цветоконтрастной линией, было предложено разработать на основе имеющихся трех датчиков света QRE1113 специальный сенсор. На вход сенсора должны поступать значения яркости от трех указанных датчиков, а на выходе он должен показывать расстояние от края цветоконтрастной линии до середины высоты равнобедренного треугольника, в вершинах которого находятся датчики света (см. рис. 1, b). Для определения положения сенсора над линией была предложена эмпирическая формула

$$Y = k \left(\frac{1}{2} S_L + S_F - \frac{1}{2} S_R \right), \quad (1)$$

где Y – расстояние от сенсора до края цветоконтрастной линии; k – эмпирический коэффициент взаимосвязи расстояния от центра сенсора до края линии и значения яркости отраженного света под ним; S_L , S_F и S_R – значения яркости отраженного света от поверхности, получаемые левым, центральным и правым датчиками света соответственно, которые выдает аналого-цифровой преобразователь бортового микроконтроллера.

Опишем предложенный алгоритм слежения за краем цветоконтрастной линии. Если $Y = 0$, то высота равнобедренного треугольника расположена вдоль края цветоконтрастной линии и корректировка не требуется (рис. 2, а). Если $Y < 0$, то во время движения мобильному роботу необходимо повернуть корпус против часовой стрелки (рис. 2, b); если $Y > 0$, то по часовой стрелке (рис. 2, c), чтобы выровнять положение и уменьшить расстояние от центра сенсора до края линии.

Было решено реализовать аппаратную часть системы управления движением мобильного робота на основе микроконтроллерной платы Arduino Mega, а программную часть – на основе библиотек пакета прикладных программ MATLAB/Simulink. В частности, библиотека Support Package for Arduino Hardware позволяет на базе созданной Simulink-модели системы управления сгенерировать управляющий код для любой из имеющихся в наборе микроконтроллерных плат семейства Arduino. Объект управления имеет два независимых ведущих колеса, соотношение скоростей вращения которых формирует требуемые линейную и угловую скорости всей платформы (рис. 3).

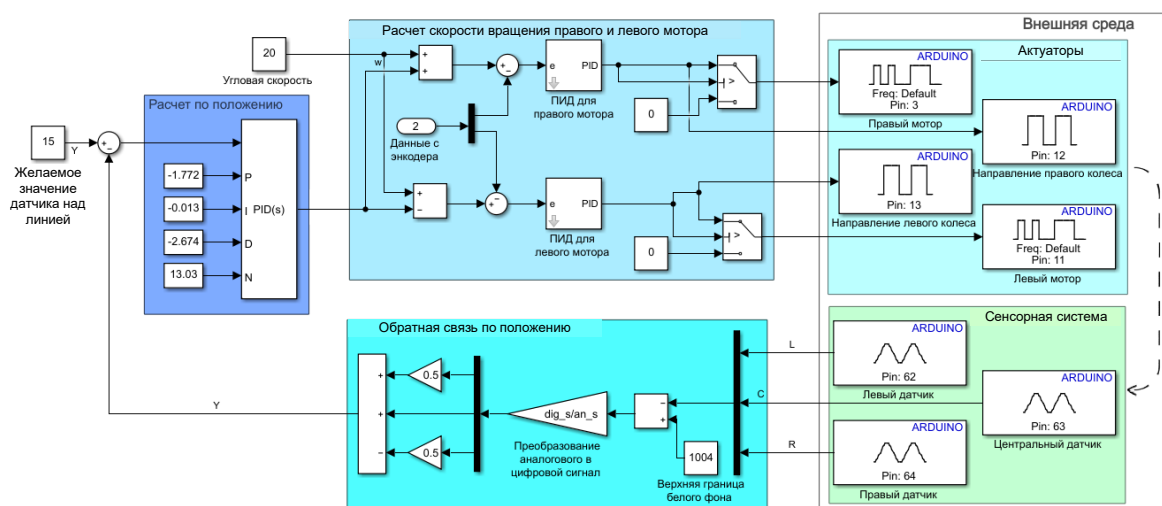


Рис. 3. Simulink-модель системы управления дифференциального привода колесной платформы с обратной связью по положению робота

Fig. 3. Simulink-model of a wheel platform differential drive control system with robot position feedback

Изначально для контроля за движением мобильного робота по линии был применен готовый ПИД-регулятор из библиотеки Control System Toolbox (URL: https://www.mathworks.com/help/pdf_doc/control/control_gs.pdf), для которого требуется ручной ввод значений коэффициентов управления: пропорционального (P), интегрального (I) и дифференциального (D) коэффициентов, а также фильтра (N) для дифференциального коэффициента. Как правило, на дифференциальный коэффициент ПИД-регулятора сильное влияние оказывает входной шум от анализируемых сенсоров. Для минимизации этого влияния применяется указанный дифференциальный фильтр. Система управления движением мобильного робота вдоль линии была разработана на основе классического ПИД-контроллера [12] с двухконтурным управлением: обратной связью по положению центра оптического сенсора над линией и обратной связью по заданной скорости вращения колес. Коэффициенты ПИД-контроллера последней связи были вычислены отдельно и в настоящей статье не рассматриваются.

Проблема настройки ПИД-регуляторов. В среде MATLAB/Simulink есть возможность использовать алгоритм автоматической настройки коэффициентов с применением дополнительного приложения PID Tuner [1, 12]. Для этого на основе реальных параметров робота, указанных ниже, в библиотеке Simscape MultiBody авторами была разработана имитационная модель мобильного робота RoboCake, которая представлена на рис. 4 в виде его отдельного блока.

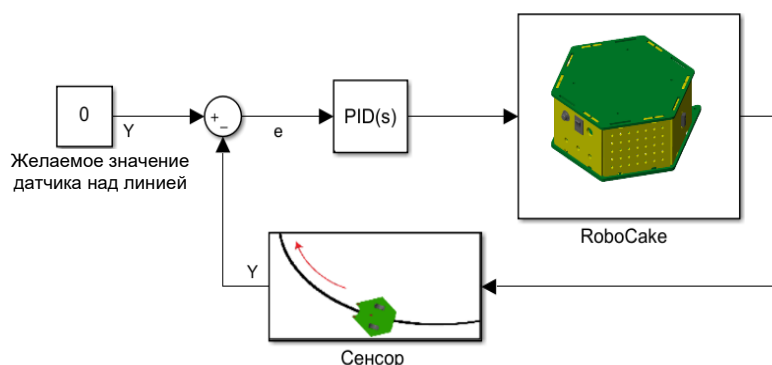


Рис. 4. Система управления имитационной моделью мобильного робота RoboCake

Fig. 4. Control system for a simulation model of a mobile robot RoboCake

Робот RoboCake имеет следующие физические параметры: радиус колеса 2,5 см, расстояние между колесами 8,5 см, ширина корпуса 16,9 см, длина корпуса 17,1 см, масса корпуса 400 г.

Для моделирования движения мобильного робота были созданы виртуальные трассы, которые представляют собой цифровые изображения цветоконтрастных линий в виде простейших геометрических фигур: окружности радиусом 1 м, эллипса с главными полуосями 0,3 и 1 м, квадрата со сторонами 1 м. На основе частей этих геометрических примитивов впоследствии можно комбинировать различные траектории, встречающиеся в реальных практических приложениях.

С целью корректного моделирования работы оптического сенсора были выбраны виртуальные трассы в виде квадрата размером 4×4 м и карта трассы, представленная в виде матрицы размером 2000×2000 пикселей (пк). С учетом этих размеров масштабный коэффициент равен 2 мм/пк, на вход каждого из трех виртуальных датчиков света подаются значения яркостей соответствующих скользящих окон в виде матриц размером 3×3 пак, ширина цветоконтрастных линий равняется 12 пак. Размер виртуальных карт взят с запасом, чтобы не стеснять движения робота и не потерять его координаты, если тот сойдет с линии.

Датчики света являются аналоговыми, поэтому в используемой авторами библиотеке Simulink Support Package for Arduino Hardware (URL: <https://www.mathworks.com/matlabcentral/fileexchange/40312-simulink-support-package-for-arduino-hardware>) цвет кодируется целыми числами от 0 до 1034. При проведении экспериментов на реальном мобильном роботе при естественном освещении и расположении датчиков на расстоянии 10 мм от поверхности трассы белый цвет кодируется числом 1004, черный – числом 15. Выходное значение находится путем вычисления среднего значения яркости всех 9 пак каждого из датчиков.

Для имитирования оптического сенсора потребовалось суммарно проанализировать фрагмент карты размером 27 пак. Сенсор считывает на карте три матрицы размером 3×3 пак, из которых формируется массив данных размером 9×3 пак элементов яркости отраженного света. Было разработано графическое окно с изображением виртуальных карт, которое позволяет видеть передвижение робота в 2D-проекции (рис. 5).

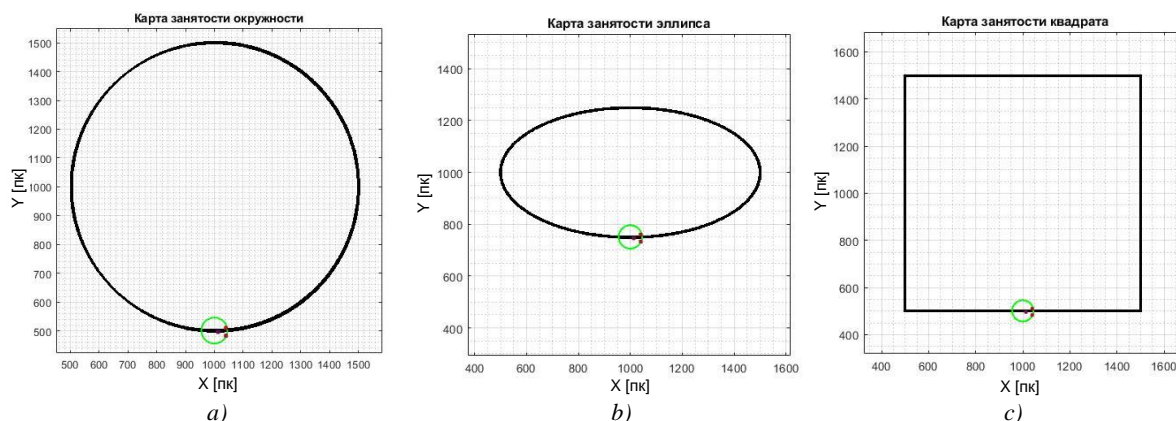


Рис. 5. Виртуальные полигоны: а) окружность ($R = 1$ м); б) эллипс ($Ra = 1$ м, $Rb = 0,25$ м); в) квадрат ($L = 1$ м)

Fig. 5. Virtual map: a) circle ($R = 1$ m); b) ellipse ($Ra = 1$ m, $Rb = 0,25$ m); c) square ($L = 1$ m)

Чтобы разработанная имитационная модель могла следовать по цветоконтрастной линии с заданной скоростью, необходимо настроить ПИД-регулятор. На первом этапе был применен метод проб и ошибок, где значения коэффициентов ПИД-регулятора подбирались вручную (табл. 1) на основе правил, указанных в работах [13, 14]. Ручным методом перебора удалось найти различные рабочие варианты наборов значений коэффициентов, при которых мобильный робот проходил всю дистанцию, однако его скорость составляла 0,3 м/с. При увеличении скорости до 1 м/с мобильный робот сходил с линии в районе фокусов указанного эллипса. Среднее время прохождения составило 210 мс. Данные результаты указывают на то, что ручная настройка ПИД-регулятора отнимает много времени и не гарантирует оптимального качества управления.

Таблица 1. Значения коэффициентов ПИД-регулятора, полученные в результате экспериментов методом ручной настройки

Table 1. The values of the PID controller coefficients obtained as a result of experiments using the manual tuning method

P	I	D	N	Время прохождения, мс Transit time, ms
2,575	-7,207	-3,675	100,000	285
3,676	-0,159	-4,665	80,250	183
-9,550	1,601	-2,827	100,000	223
6,252	-4,969	1,045	20,610	192
-4,047	-0,005	-4,085	1,280	167
Среднее значение				210

Следующим этапом была автоматическая настройка ПИД-регулятора с помощью приложения PID Tuner (URL: <https://www.mathworks.com/help/control/ref/pidtuner-app.html>), входящего в библиотеку System Identification Toolbox. В инструкции к PID Tuner указано, что данный инструмент может выполнять свои функции автоматической настройки только при условии линеаризованной модели объекта управления. Однако в связи с тем, что в разработанной имитационной модели учитываются трение и инерция, она является изначально нелинейной. Поэтому авторам пришлось воспользоваться дополнительной возможностью приложения PID Tuner, которое численным методом подбирает исследуемому объекту максимально правдоподобную передаточную функцию. Данный метод включает следующие этапы: 1) воздействие на объект управления тестовыми входными данными; 2) сбор полученных выходных данных, на основе которых происходит идентификация самого объекта управления; 3) обработку полученных данных отклика; 4) корректировку структуры и параметров новой модели объекта управления;

5) сохранение модифицированного объекта управления. После этого PID Tuner выдает эталонные значения, которые отображают новый отклик с обратной связью. Полученные данные были применены в готовой модели (табл. 2). С помощью автоматической настройки удалось уменьшить среднее время прохождения дистанции до 133 мс, однако не удалось достичь требуемой скорости в 1 м/с. При этом даже со скоростью 0,45 м/с мобильный робот двигался рывками, поэтому было решено использовать генетический алгоритм.

Таблица 2. Значения коэффициентов ПИД-регулятора, полученные в результате автоматической настройки в приложении PID Tuner

Table 2. The values of the PID controller coefficients obtained as a result of automatic tuning by the PID Tuner application

P	I	D	N	Время прохождения, мс <i>Transit time, ms</i>
0,067	0,006	0,648	1,280	126
0,001	1,551	0,000	0,152	142
-0,012	-5,870	-0,473	0,129	138
0,572	0,001	0,005	1,540	122
0,383	0,010	0,035	2,550	137
<i>Среднее значение</i>				133

Оптимизация значений параметров ПИД-регулятора на основе разработанного генетического алгоритма. В связи с тем что настройку ПИД-регулятора можно представить как задачу поиска локального минимума для функции четырех переменных, было предложено воспользоваться одним из современных методов эволюционной оптимизации – генетическим алгоритмом. Основным преимуществом генетического алгоритма, который оказался эффективным для настройки ПИД-регулятора, является то, что процесс поиска локального минимума функции пригодности реализуется параллельно из большого числа точек фазового пространства управляющих коэффициентов ПИД-регулятора.

Экспериментальные исследования проводились в соответствии со следующими шагами генетического алгоритма (рис. 6), реализованного в библиотеке Global Optimization Toolbox (URL: <https://www.mathworks.com/help/gads/genetic-algorithm-options.html>):

1. Подбор коэффициентов для каждой особи из популяции.
2. Загрузка данных в имитационную модель, где время симуляции занимает 6 с и за это время мобильный робот совершает полный круг по эллиптической кривой.
3. Формирование выходных данных в виде матрицы и графиков.
4. Вычисление функции пригодности для эллиптической кривой, где вычисляется расстояние e .
5. Проверка приведенных ниже критериев останковки:

MaxGenerations – алгоритм останавливается, если было достигнуто максимальное число итераций.

MaxStallGenerations – алгоритм останавливается, если среднее относительное изменение в значении наилучшей функции пригодности **MaxStallGenerations** меньше или равно **FunctionTolerance**, где для последней функции устанавливается предел, который не должен превышать среднее изменение функции пригодности.

MaxTime – превышение заданного времени останковки.

6. Если условия не выполняются, то следует вернуться к шагу 1 и сформировать следующее поколение.

Более подробное описание работы предложенного генетического алгоритма можно найти по ссылке на [GitHub.com](https://github.com/TatyanaKim3160/RoboCake/blob/master/блок-схема.docx) (URL: <https://github.com/TatyanaKim3160/RoboCake/blob/master/блок-схема.docx>).

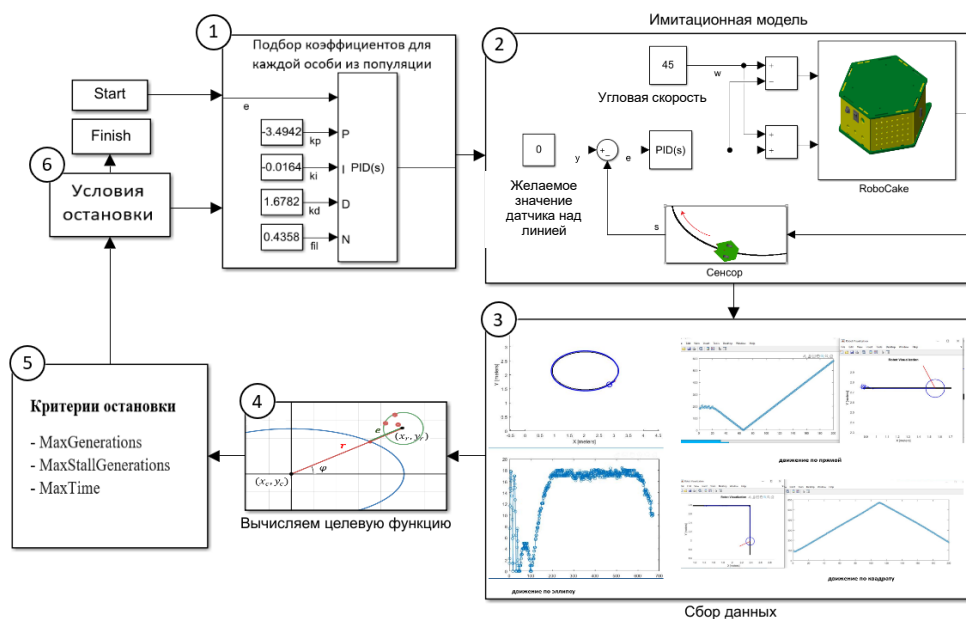


Рис. 6. Схема работы генетического алгоритма, позволяющего найти требуемые коэффициенты для ПИД-регулятора

Fig. 6. The scheme of the genetic algorithm, which allows to find the required coefficients for the PID controller

Формализация задачи и формирование функции пригодности для разработанного генетического алгоритма. Первым и наиболее важным шагом при использовании генетических алгоритмов является формализация задачи. Авторы предложили, чтобы популяция в каждом поколении (конкретные значения популяции в каждой итерации) была представлена в виде набора особей, где каждая особь кодируется одной хромосомой в виде вектора из коэффициентов P_i, I_i, D_i и N_i :

$$\underbrace{P_1 \ I_1 \ D_1 \ N_1}_{\text{Особь 1}} \ \underbrace{P_2 \ I_2 \ D_2 \ N_2}_{\text{Особь 2}} \ \underbrace{\dots \ \dots \ \dots \ \dots}_{\dots} \ \underbrace{P_n \ I_n \ D_n \ N_n}_{\text{Особь } n} \ . \quad (2)$$

Другими словами, в рассматриваемом случае особь содержит одну хромосому, в которой присутствуют четыре гена, отвечающих за пропорциональный, интегральный и дифференциальный коэффициенты настраиваемого ПИД-регулятора, а также коэффициент фильтра для дифференциальной составляющей регулятора. Подобная задача формализации уже использовалась в работе [15]. В настоящей статье для решения задачи оптимизации коэффициентов ПИД-регулятора системы управления движением мобильного робота предложена собственная функция пригодности.

Вторым шагом является разработка функции пригодности, минимизирующей расстояния между центром сенсора мобильного робота и краем цветоконтрастной линии. Операция оценивания функции пригодности является основополагающей для эффективной работы генетического алгоритма. Для успешного формирования функции пригодности необходимо определить условия, при которых в процессе движения мобильного робота будут подбираться коэффициенты для ПИД-регулятора, позволяющие роботу в процессе движения минимизировать расстояние до края цветоконтрастной линии (рис. 7).

Рассмотрим принцип формирования функции пригодности на примере первого виртуального полигона, в который поочередно загружались две замкнутые кривые в виде эллипсов с параметрами $Ra = 1$ м, $Rb = 0,5$ м и $Ra = 1$ м, $Rb = 0,25$ м. Имитационная модель мобильного робота двигалась со скоростью 1 м/с. Функция пригодности $F_{ГА}$ заключается в том, чтобы за выделенное модельное время робот смог проехать вдоль замкнутой линии максимально возможное рас-

стояние L . Суммарная ошибка E , являющаяся расстоянием между центром робота и краем линии, должна быть минимальной. В результате функция пригодности для каждой особи вычисляется следующим образом:

$$F_{\Gamma A} = \frac{E}{L}. \quad (3)$$

Путь, пройденный роботом за выделенное модельное время, определяется равенством

$$L = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}, \quad (4)$$

где x_i и y_i – координаты положения центра мобильного робота на i -м шаге моделирования, n – количество шагов модельного времени.

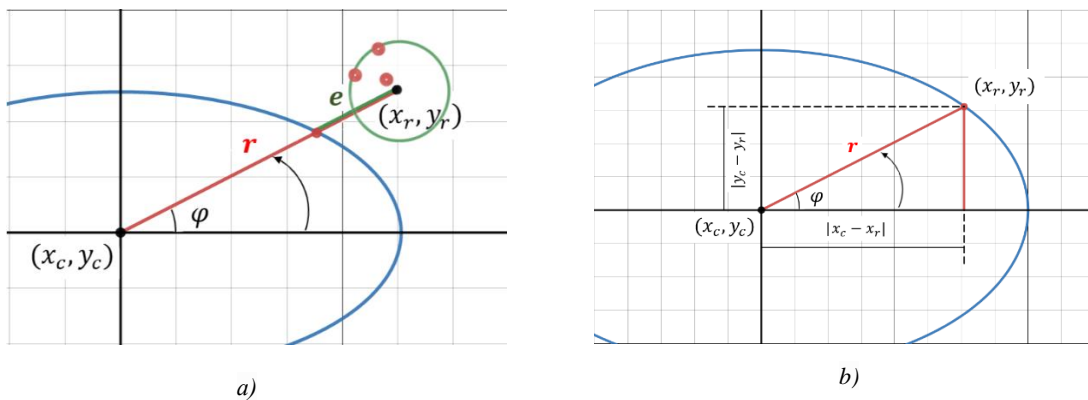


Рис. 7. Графики, демонстрирующие формирование функции пригодности: а) определение ближайшего расстояния e от центра робота до цветоконтрастной линии; б) вычисление угла φ пройденного расстояния между осью абсцисс и текущим положением робота с координатами x_r, y_r

Fig. 7. Graphs demonstrating the formation of the fitness function: a) determination of the closest distance e from the center of the robot to the color contrast line; b) calculating the angle φ of the distance traveled between the abscissa axis and the current coordinates of the robot x_r, y_r ;
b) determination of the closest distance e from the center of the robot with coordinates x_r, y_r

Будем считать, что ошибка рассогласования e движения робота вдоль края линии равняется расстоянию между текущим положением робота и точкой эллипса, получаемой пересечением края линии с отрезком, который соединяет центры эллипса и робота (рис. 7, а):

$$e = \sqrt{(x_c - x_r)^2 + (y_c - y_r)^2} - r, \quad (5)$$

где x_c и y_c – координаты центра эллипса; x_r и y_r – текущие координаты положения центра мобильного робота; r – радиус эллипса, соответствующий пройденному роботом углу φ :

$$\varphi = \arcsin \frac{|y_c - y_r|}{\sqrt{(x_c - x_r)^2 + (y_c - y_r)^2}}; \quad (6)$$

$$r = \frac{ab}{\sqrt{b^2 \cos^2 \varphi + a^2 \sin^2 \varphi}}. \quad (7)$$

Тогда суммарная ошибка E , накопленная за время прохождения роботом пути, будет равняться сумме квадратов минимального расстояния между роботом и линией:

$$E = \sum_{i=1}^n e_i^2. \quad (8)$$

Рассматриваемый подход использовался также для виртуальных полигонов «окружность» и «квадрат», описания их функций пригодности в статье не приводятся. При этом во время работы генетического алгоритма были получены промежуточные управляющие операторы.

Создание начальной популяции и выбор операторов для предложенного генетического алгоритма. Для формирования начальной популяции было проведено более 100 пробных запусков, где количество особей располагалось в диапазоне от 5 до 150. Были выбраны оптимальные управляющие операторы генетического алгоритма с минимальной ошибкой отклонения. Не всегда большое количество особей в одной итерации приводит к хорошим результатам, наиболее эффективное число особей для используемой модели – от 25 до 50 на популяцию.

Для того чтобы сократить время поиска значений коэффициентов ПИД-регулятора и, следовательно, сузить пространство поиска, были найдены границы области допустимых значений (см. табл. 2), которые использовались при последующей работе генетического алгоритма.

Следующим этапом для реализации предложенного генетического алгоритма является выбор операторов для правильной оптимизации ПИД-регулятора из библиотеки Optimization Toolbox (URL: https://www.mathworks.com/help/pdf_doc/optim/optim.pdf). С целью получения качественных результатов и повышения скорости обучения работа по восьми возможным параметрам настройки генетического алгоритма путем перебора и сравнения были выбраны:

1. *Оператор селекции.* Функция селекции предназначена для отбора родительских хромосом в новое поколение. Для этого оказался более эффективным метод стохастической равномерной селекции, используемый по умолчанию. Суть метода заключается в том, что родительские хромосомы могут выбираться более одного раза и передаваться нескольким потомкам.

2. *Оператор скрещивания.* В процессе скрещивания формируется следующее поколение. Из семи предложенных в библиотеке параметров путем перебора была выбрана функция арифметического пересечения Arithmetic, которая использует среднеарифметическое взвешенное значение от предыдущего поколения.

3. *Оператор мутации.* Применяется для случайно выбранной особи, у которой один из случайно выбранных генов меняется на случайное число. Значение этого числа устанавливается методом Гаусса.

4. *Оператор элитного отбора.* После одного цикла итерации идет отбор элитных особей. Подсчетом элиты определяется число лучших особей, которые гарантированно переходят в следующее поколение без изменений. Их количество должно быть меньше либо равно количеству особей в популяции. В проведенных экспериментах количество элитных особей составило 4 % от всего количества особей.

Выбранные операторы и значения приведены в табл. 3. Вычисленные значения параметров подробно описаны в справочной системе пакета MATLAB R2020b (URL: <https://www.mathworks.com/help/gads/genetic-algorithm-options.html>).

Таблица 3. Операторы и значения параметров для генетического алгоритма

Table 3. Operators and parameter values for genetic algorithm

Операторы <i>Operators</i>	Значения операторов <i>Values of operators</i>
Population type	50
Creation function	Uniform
Scaling function	Rank
Selection function	Stochastic uniform
Mutation function	Gaussian
Elite count	2
Crossover function	Arithmetic
Crossover fraction	0,8
Use parallel	1
Max Generations	25
Upper bounds	5·[1 1 1 1]
Lower bounds	-5·[1 1 1 0]

Было проведено около 80 экспериментов, и только в пяти из них (табл. 4) выполнялись условия, заданные с помощью дополнительной библиотеки пакета MATLAB R2020b (табл. 3). Среднее значение времени прохождения робота по кривой эллипса сократилось до 56 мс, причем найденные значения коэффициентов ПИД-регулятора позволяют роботу двигаться как по эллипсу, так и по окружности.

Таблица 4. Значения коэффициентов ПИД-регулятора, которые прошли отбор и удовлетворяют всем критериям пригодности

Table 4. The values of the PID controller coefficients that have passed the selection and meet all fitness criteria

P	I	D	N	Значение минимальной функции пригодности <i>The value of the minimum fitness function</i>	Пройденное время, мс <i>Elapsed time, ms</i>
-3,494	-0,016	1,678	0,436	1,820	61
-2,903	-0,368	-0,716	-0,461	0,001	52
2,001	-0,965	-3,932	3,920	0,003	54
-4,627	0,150	0,301	4,948	0,126	55
-1,000	0,000	-0,062	1,199	0,452	58
<i>Среднее значение</i>					56

За 25 поколений отбора (рис. 8) были найдены значения операторов (см. табл. 3), соответствующие минимуму функции пригодности: $F_{GA} = 1,820$ (лучшее значение). При этом среднее значение функции пригодности для сгенерированной случайным образом начальной популяции решений по всем особям поколения составило 4,523.



Рис. 8. Процесс оптимизации искоемых коэффициентов с помощью генетического алгоритма

Fig. 8. Optimization of the desired coefficients using a genetic algorithm

Использованные методы настройки для разработанного генетического алгоритма доказали свою эффективность в выборе лучших результатов за счет повышения устойчивости характеристик и показателей производительности.

Пример движения робота по кривой эллипса с параметрами $Ra = 1$ и $Rb = 0,5$ м и полученными значениями коэффициентов ПИД-регулятора приведен на рис. 9, где прерывистой линией обозначено желаемое движение робота относительно угла его положения с линейной скоростью 1 м/с, а непрерывной линией показан график функции изменения радиуса движения имитационной модели. На рис. 9, а видно, что многократный запуск с максимальным количеством особей и итераций в некоторых случаях может привести к ухудшению эффективности работы коэффициентов.

На рис. 10 представлен график накопления ошибки E в процессе движения робота по кривой эллипса со скоростью 1 м/с.

Во время исследований было осуществлено более 80 симуляций, благодаря которым были определены подходящие количество особей в популяции и количество итераций для данного робота. Задание дополнительных параметров для генетического алгоритма требует опыта и знаний, так как их применение сокращает время обучения и улучшает результат.

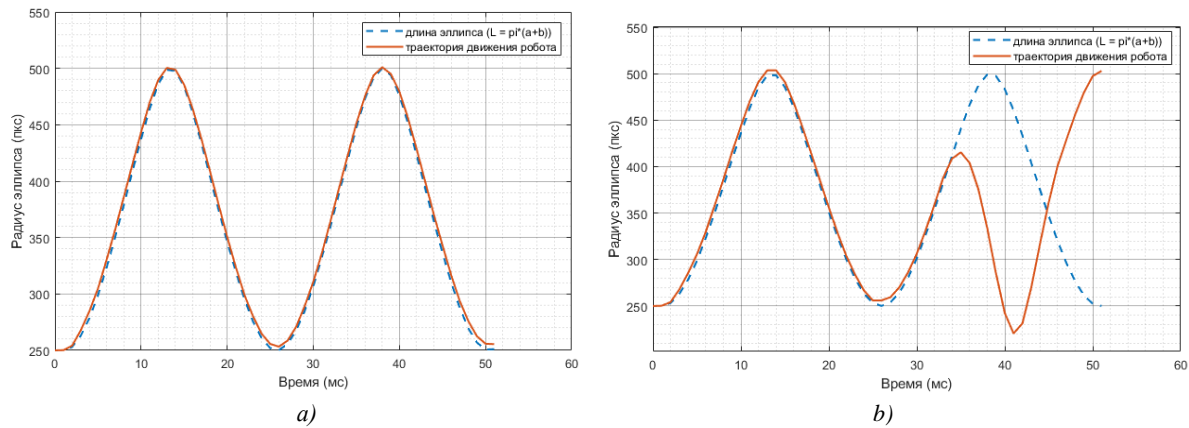


Рис. 9. Результаты запуска генетического алгоритма для управления движением робота по кривой эллипса в течение 52 мс модельного времени: а) итераций 25, особей 50; б) итераций 75, особей 70

Fig. 9. Results of running a genetic algorithm to control the robot's motion along an ellipse curve for 52 ms of model time: a) iterations 25, individuals 50; b) iterations 75, individuals 70

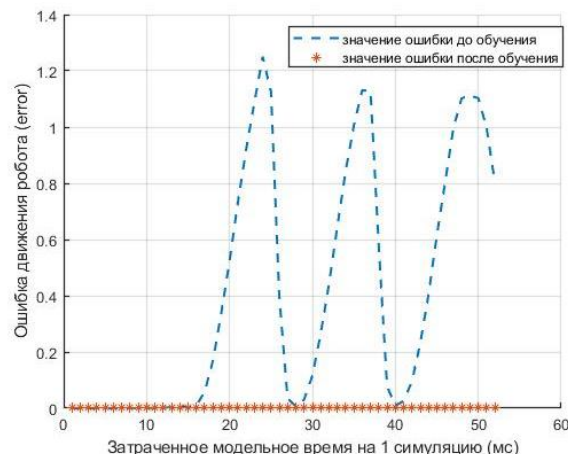


Рис. 10. График ошибки движения робота по кривой эллипса

Fig. 10. Graph of the error of the robot's movement along the ellipse curve

Предложенная функция пригодности, описываемая формулами (3)–(8), и выбранные параметры настройки генетического алгоритма позволили найти значения коэффициентов ПИД-регулятора, обеспечивающего приемлемое управление движением робота по цветоконтрастной линии. Моделирование проводилось с применением метода Rapid Accelerator на шестиядерном процессоре Intel(R) Core (TM) i5-9600KF с 32 ГБ оперативной памяти для хранения популяций решений. Было затрачено 6 ч рабочего времени.

Заключение. В статье описан процесс создания системы управления движением мобильного робота RoboCake Beginner по цветоконтрастной линии. Для этого была разработана и реализована система управления с обратной связью от оптического сенсора положения, которая способна управлять как реальным роботом, так и его имитационной моделью. Имитационная модель робота способна двигаться со скоростью 1 м/с по окружности с радиусом 1 м и эллипсу с радиусами 1 и 0,5 м. На более сложном полигоне в форме эллипса с радиусами 1 и 0,25 м мобильный робот может двигаться со скоростью 0,8 м/с. Полученные коэффициенты

ПИД-регулятора были использованы в системе управления роботом RoboCake Beginner, который двигался со скоростью 0,4 м/с. Низкая скорость движения реального робота обуславливается недостаточно мощными двигателями, которые в нем использовались.

Для повышения качества движения мобильного робота первоначально были применены два метода настройки ПИД-регулятора: ручной и автоматический (с помощью приложения PID Tuner). В процессе усложнения траекторий движения робота был выявлен ряд недостатков, таких как перерегулирование объекта управления, потеря полезного времени для подбора коэффициентов и перезапуск модели, несоответствие точной динамики робота. Это дало основание для применения генетического алгоритма на следующем этапе решения задачи оптимизации коэффициентов ПИД-регулятора.

В работе предложен подход к настройке решателя, позволяющий уменьшить вычислительные затраты непрерывного моделирования в процессе эволюционного поиска с помощью генетического алгоритма. Были подобраны параметры для операторов популяции, селекции, скрещивания и мутации. Для оптимизации коэффициентов ПИД-регулятора разработана функция пригодности. На основе полученных управляющих коэффициентов учебный мобильный робот способен двигаться вдоль нелинейной цветоконтрастной линии, минимизируя расстояние между центром оптического сенсора и краем линии.

Применение генетического алгоритма для учебного робота позволило получить приемлемые результаты благодаря использованию предложенных функций пригодности для каждой карты (замкнутой трассы в виде окружности, эллипса и квадрата) отдельно и выбору подходящих параметров генетических операторов.

В результате анализа большого количества проведенных численных экспериментов было найдено такое количество популяций и итераций, которое позволило минимизировать вычислительные затраты. Была установлена взаимосвязь между количеством запусков популяций и числом итераций, а также факт минимального влияния значений интегрального коэффициента в ПИД-регуляторе. В связи с этим рекомендуется упростить систему управления до ПД-регулятора.

Мобильные роботы находят все больше практического применения в автоматизации производства, сфере обслуживания и других непромышленных направлениях. Опыт, полученный при проведении экспериментов по движению мобильного робота RoboCake Beginner по контрастной линии, может быть использован для построения более сложных роботизированных систем, что значительно снизит риски на начальной стадии их проектирования.

Вклад авторов. Т. Ю. Ким разработала программный комплекс по настройке ПИД-регулятора, провела экспериментальные исследования с ручной и полуавтоматической настройкой ПИД-регулятора, программно реализовала функцию пригодности для генетического алгоритма, а также настроила его параметры. Г. А. Прокопович поставил задачи, определил план и обосновал актуальность работы, предложил формализацию основных компонентов решения поставленной задачи с помощью генетического алгоритма.

Список использованных источников

1. Автоматизированная настройка ПИД-регулятора для объекта управления следящей системы с использованием программного пакета MATLAB Simulink / А. В. Филиппов [и др.] // Наука, техника и образование. – 2015. – № 12(18). – С. 53–59.
2. Aström, K. J. The future of PID control / K. J. Aström, T. Hägglund // Control Engineering Practice. – 2001. – Vol. 9, no. 11. – P. 1163–1175. [https://doi.org/10.1016/S0967-0661\(01\)00062-4](https://doi.org/10.1016/S0967-0661(01)00062-4)
3. Иващенко, Н. Н. Автоматическое регулирование: теория и элементы систем / Н. Н. Иващенко. – М. : Машиностроение, 1978. – 737 с.
4. Martins, F. G. Tuning PID controllers using the ITAE criterion / F. G. Martins // Intern. J. of Engineering Education. – 2005. – Vol. 21, no. 5. – P. 867–873.
5. Kose, O. PID controlled line follower robot design on indoor 3D networks / O. Kose, I. R. Karas // IAES Intern. Conf. on Electrical Engineering, Computer Science and Informatics (EECSI-2014). – Yogyakarta, Indonesia, 2014. – P. 20–21.

6. Варламов, И. PID-EXPERT – автоматизация автоматизации / И. Варламов, П. Зубов // *Control Engineering*. – 2017. – № 2(68). – С. 98–101.
7. Vladu, E. E. Controller tuning using genetic algorithms [Electronic resource] / E. E. Vladu, T. L. Dragomir // 1st Romanian-Hungarian Joint Symp. on Applied Computational Intelligence (SACI 2004). – Springer, 2004. – Mode of access: https://www.researchgate.net/publication/228954545_Controller_Tuning_Using_Genetic_Algorithms. – Date of access: 12.05.2021.
8. Genetic algorithm / M. Ünal [et al.] // *Optimization of PID Controllers Using Ant Colony and Genetic Algorithms*. – Berlin : Springer, 2013. – 449 p. https://doi.org/10.1007/978-3-642-32900-5_3
9. Саймон, Д. Алгоритмы эволюционной оптимизации. Биологически обусловленные и популяционно-ориентированные подходы к компьютерному интеллекту / Д. Саймон ; пер. с англ. А. В. Логунова. – М. : ДМК Пресс, 2020. – 1002 с.
10. Разработка генетического алгоритма с адаптивными мутациями для определения глобального экстремума функции n -переменных [Электронный ресурс] / А. Н. Кошев [и др.] // Интернет-журнал «Науковедение». – 2016. – № 8(6). – С. 1–13. – Режим доступа: <http://naukovedenie.ru/PDF/32TVN616.pdf>. – Дата доступа: 12.05.2021.
11. Подлазова, А. В. Генетические алгоритмы на примерах решения задач раскроя / А. В. Подлазова // *Проблемы управления*. – 2008. – № 2. – С. 57–63.
12. Koo, Y. C. Motor speed controller for differential wheeled mobile robot / Y. C. Koo, E. A. Bakar // *ARPN J. of Engineering and Applied Sciences*. – 2015. – Vol. 10, no. 22. – P. 10698–10702.
13. PID Control / ed. W. S. Levine // *The Control Handbook*. – Piscataway, NJ : IEEE Press, 1996. – P. 198–209.
14. O'Dwyer, A. Handbook of PI and PID Controller Tuning Rules / A. O'Dwyer. – London : Imperial College Press, 2003. – 564 p.
15. Mirzal, A. PID parameters optimization by using genetic algorithm / A. Mirzal, Sh. Yoshii, M. Furukawa // *ISTECS J.* – 2006. – Vol. 8, no. 11. – P. 34–43.

References

1. Filippov A. V., Kosolapov M. A., Maslov I. A. *Automated tuning of the PID controller for the control object of the servo system using the MATLAB Simulink software package*. Nauka, tehnika i obrazovanie [Science, Technology and Education], 2015, no. 12(18), pp. 53–59 (In Russ.).
2. Aström K. J., Hägglund T. The future of PID control. *Control Engineering Practice*, 2001, vol. 9, no. 11, pp. 1163–1175. [https://doi.org/10.1016/S0967-0661\(01\)00062-4](https://doi.org/10.1016/S0967-0661(01)00062-4)
3. Ivaschenko N. N. Avtomaticheskoe regulirovanie: teoriya i elementi system. *Automatic Regulation: Theory and Elements of Systems*. Moscow, Mashinostroenie, 1978, 737 p. (In Russ.).
4. Martins F. G. Tuning PID controllers using the ITAE criterion. *International Journal of Engineering Education*, 2005, vol. 21, no. 5, pp. 867–873.
5. Kose O., Karas I. R. PID controlled line follower robot design on indoor 3D networks. *IAES International Conference on Electrical Engineering, Computer Science and Informatics (EECSI-2014)*. Yogyakarta, Indonesia, 2014, pp. 20–21.
6. Varlamov I., Zubov P. PID-EXPERT – automation automation's. *Control Engineering*, 2017, no. 2(68), pp. 98–101 (In Russ.).
7. Vladu E. E., Dragomir T. L. Controller tuning using genetic algorithms. *1st Romanian-Hungarian Joint Symposium on Applied Computational Intelligence (SACI 2004)*. Springer, 2004. Available at: https://www.researchgate.net/publication/228954545_Controller_Tuning_Using_Genetic_Algorithms (accessed 12.05.2021).
8. Ünal M., Ak A., Topuz V., Erdal H. Genetic algorithm. *Optimization of PID Controllers Using Ant Colony and Genetic Algorithms*. Berlin, Springer, 2013, 449 p. https://doi.org/10.1007/978-3-642-32900-5_3
9. Simon D. *Evolutionary Optimization Algorithms*. John Wiley & Sons, 2013, 784 p.
10. Koshev A. N., Salmin V. V., Generalova A. A., Bichkov D. S. *The development of genetic algorithm with adaptive mutations to determine the global extremum function of n-variables*. Internet-jurnal "Naukovedenie" [Online Magazine "Science"], 2016, vol. 8, no. 6, pp. 1–13. Available at: <http://naukovedenie.ru/PDF/32TVN616.pdf> (accessed 12.05.2021).
11. Podlazov A. V. *Genetic algorithms based on examples of solving cutting problems*. Problemi upravleniya [Management Problems], 2008, vol. 2, pp. 57–63 (In Russ.).
12. Koo Y. C., Bakar E. A. Motor speed controller for differential wheeled mobile robot. *ARPN Journal of Engineering and Applied Sciences*, 2015, vol. 10, no. 22, pp. 10698–10702.

13. PID control. In W. S. Levine (ed.). *The Control Handbook*. Piscataway, New Jersey, IEEE Press, 1996, pp. 198–209.
14. O'Dwyer A. *Handbook of PI and PID Controller Tuning Rules*. London, Imperial College Press, 2003, 564 p.
15. Mirzal A., Yoshii Sh., Furukawa M. PID parameters optimization by using genetic algorithm. *ISTECS Journal*, 2006, vol. 8, no. 11, p. 34–43.

Информация об авторах

Ким Татьяна Юрьевна, аспирант, Объединенный институт проблем информатики Национальной академии наук Беларуси.

E-mail: tatyana_kim92@mail.ru

Прокопович Григорий Александрович, кандидат технических наук, доцент, заведующий лабораторией робототехнических систем, Объединенный институт проблем информатики Национальной академии наук Беларуси.

E-mail: rprakapovich@robotics.by

Information about the authors

Tatyana Yu. Kim, Postgraduate Student, The United Institute of Informatics Problems of the National Academy of Sciences of Belarus.

E-mail: tatyana_kim92@mail.ru

Ryhor A. Prakapovich, Cand. Sci. (Eng.), Associate Professor, Head of the Robotic Systems Laboratory, The United Institute of Informatics Problems of the National Academy of Sciences of Belarus.

E-mail: rprakapovich@robotics.by