

СИДОРЕНКО А.В., СОЛОДУХО Н.А.

АЛГОРИТМ ОГИБАНИЯ ПРЕПЯТСТВИЙ ПРИ ПЕРЕМЕЩЕНИИ МОБИЛЬНОГО РОБОТА

Факультет радиофизики и компьютерных технологий
Белорусского государственного университета
г. Минск, Республика Беларусь

Предложен алгоритм и разработана компьютерная программа для моделирования огибания препятствий роботизированной системой. Для расширения возможностей перемещений роботизированной системой алгоритм совмещает основные элементы алгоритма *vfh* и разработанного нами ранее алгоритма I [1] и позиционируется в данной работе как алгоритм II. Проведена оценка среднего времени огибания препятствий различного типа, включая пять препятствий кубического типа, препятствие в виде длинной стены и сложное препятствие. В процессе вычислительного эксперимента с использованием симуляционного пространства Gazebo II рассчитаны временные параметры перемещения роботом препятствий различного типа, включая пять препятствий кубического типа, препятствие в виде стены и сложное препятствие. В процесс вычислительного эксперимента проведена статистическая обработка полученных результатов. Показано, что при использовании предложенного алгоритма оценка среднего времени огибания препятствий роботизированной системой снижается в отдельных случаях в 7,2 раза по сравнению с использованием алгоритма I.

Ключевые слова: робот, перемещение, препятствия, ROS 2, Gazebo II

Введение

С учетом недавних достижений в области искусственного интеллекта и развития беспилотных летательных аппаратов при перемещении в двумерном и трехмерном пространствах, возникает задача обхода препятствий для роботизированных систем.

В работе [1], где реализован наш алгоритм I для обхода препятствий, несмотря на отсутствие методов машинного обучения, что могло бы замедлить процесс и увеличить вычислительную нагрузку, движение робота осуществляется не по кратчайшему маршруту. Алгоритм *vfh*, описанный в работе [2], демонстрирует оптимальные траектории при обходе множества препятствий. Тем не менее, его эффективность снижается, когда речь идет о длинных препятствиях, расположенных перпендикулярно траектории робота на пути к цели. В таких случаях алгоритм может сталкиваться с трудностями в планировании обхода, что приводит к необходимости использования дополнительных стратегий или модификаций алгоритма для более успешного маневрирования вокруг подобных препятствий.

В данной работе мы предлагаем новый алгоритм II, в котором мы интегрировали алгоритм *vfh* и алгоритм I. Такая комбинация позволяет использовать сильные стороны обоих подходов: алгоритм *vfh* обеспечивает быструю реакцию на изменения в окружающей среде и эффективное обхождение препятствий, в то время как алгоритм I отвечает

за более безопасное и адаптивное маневрирование. Это сочетание позволит роботу не только находить оптимальные маршруты, но и эффективно справляться с длинными и сложными препятствиями, которые могут встречаться на его пути. Объединенный алгоритм II нацелен на улучшение общей производительности навигации и адаптацию к динамическим условиям окружающей среды. Изначально робот начинает движение, применяя алгоритм *vfh*. При наличии в процессе перемещения робота длинного препятствия, расположенного перпендикулярно цели движения робота, в компьютерной программе, описывающей алгоритм, происходит переключение на использование алгоритма I. Работа является продолжением работы, представленной в [1, 3].

Проведение вычислительного эксперимента

Программно реализованные алгоритмы, использованные в нашей разработанной модели управления системой мобильных роботов, сделали возможным проведение вычислительного эксперимента с помощью созданной компьютерной программы.

Структурная схема предложенного нами алгоритма приведена на рисунке 1.

При работе с симуляцией движения робота использовался визуализатор Gazebo II и ROS 2 Humble Hawksbill. Для управления роботом применялись команды командной оболочки UNIX bash в ОС Linux Ubuntu 22.04 Jammy Jellyfish. Исходный код представлен в [4] в ветке “*vfh-modified-by-mikkita*”.

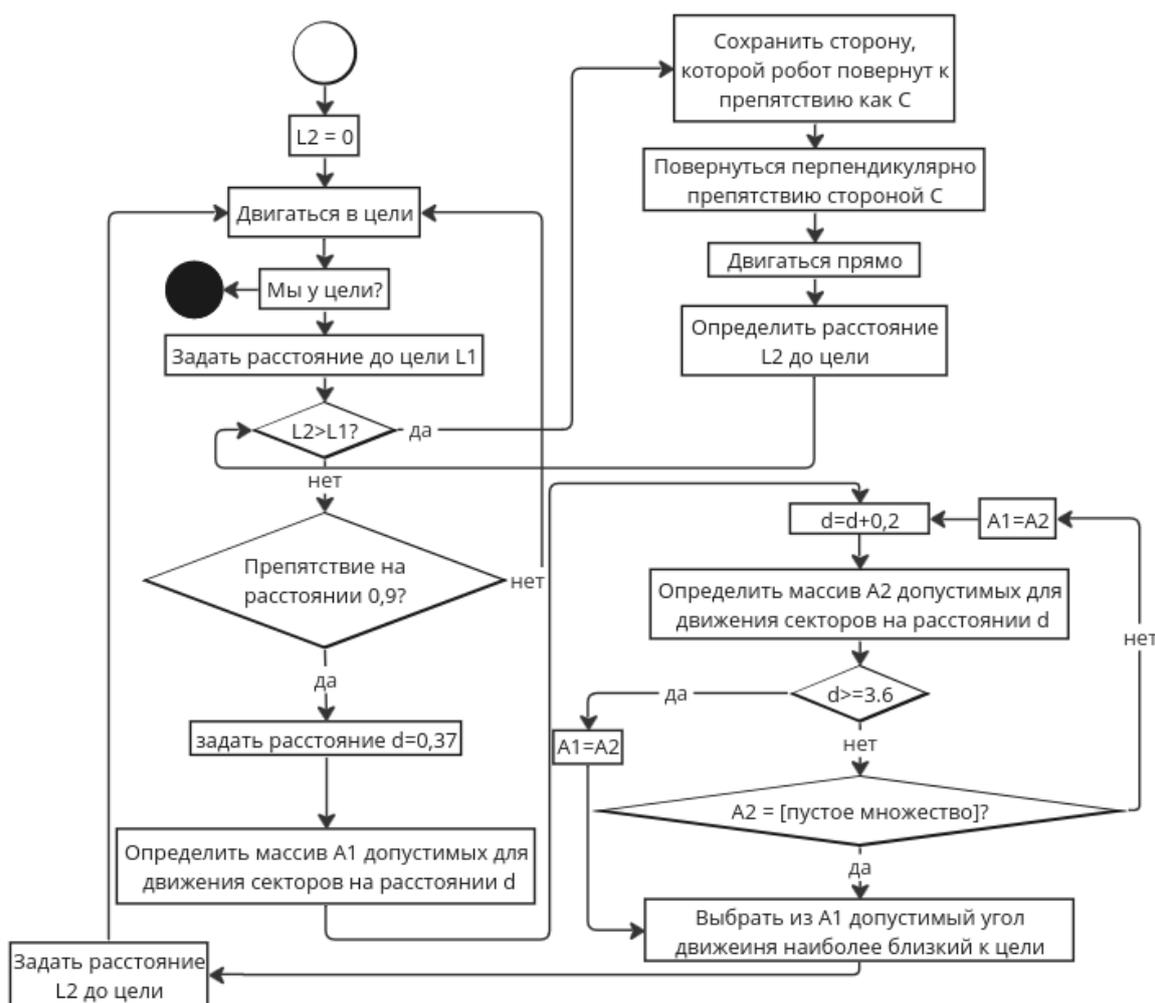


Рисунок 1. Структурная схема предложенного алгоритма II

Для работы алгоритма следует запустить скрипт `bash lines.sh`. Для задания конечной точки движения необходимо задать координаты X и Y (переменная `targetX` и переменная `targetY`, соответственно) в строках 3-16 файла `lines.sh`.

Модификация алгоритма для огибания препятствий

При моделировании системы перемещения робота в качестве среды была использована симуляционная поверхность размером 20 на 20 клеток, на которой размещены различные препятствия: одно препятствие кубического типа (рисунок 2а); два препятствия кубического типа (рисунок 2б); три препятствия кубического типа (рисунок 2в); четыре препятствия кубического типа (рисунок 2г); пять препятствий кубического типа (рисунок 2д); препятствие в виде длинной стены (рисунок 2е); сложное препятствие (рисунок 2ж). Стартовая точка при описании движения препятствий кубического типа в координатах $(-2,0)$, конечная точка имеет координаты $(9,0)$. Применение длинной стены и

сложного препятствия имеет при расчетах координаты стартовой точки $(-3,5)$ и $(-3,5)$, соответственно, а координаты конечной точки составляют: $(4,3)$ и $(4,3)$.

Алгоритм `vfn` основывается на определении свободного сектора, в котором нет препятствий на требуемом расстоянии и который максимально близок к цели. Используя данные со встроенных датчиков робота, выбирается такое направление движения, чтобы избежать столкновений с задетектированными препятствиями и при этом двигаться, приближаясь к цели.

Далее мы более подробно рассмотрим работу алгоритма `vfn` применительно к нашей программе. Этот алгоритм позволяет роботу эффективно обходить препятствия, анализируя окружение и выбирая оптимальные направления для движения. Мы обсудим ключевые аспекты реализации алгоритма, его параметры, а также способы адаптации к различным условиям окружающей среды. Когда робот начинает движение и его расположение приближается к препятствию на расстояние около 0,89 единицы координатной сетки, запускается цикл оценки доступных для движения секторов. В рамках этого

цикла происходит анализ окружающей среды, где каждый сектор оценки находит свое соответствие с данными о препятствиях. Робот оценивает, какие направления являются безопасными для движения и выбирает наиболее подходящий сектор для дальнейшего перемещения. Этот процесс происходит, когда расстояние до препятствия составляет 0,89 единицы координатной сетки. На данном этапе робот анализирует данные о расположении препятствий, определяя, какие сектора доступны для движения. Путем оценки расстояний и углов к каждому сектору, робот может с точностью определить наиболее безопасные и оптимальные направления для дальнейшего перемещения, минимизируя риск столкновения. Запуск цикла оценки доступных секторов осуществляется поэтапно. В каждой итерации расстояние, на котором проводится оценка, увеличивается на 0,2 единицы координатной сетки. Это позволяет роботу постепенно расширять область анализа, начиная с ближайших секторов и постепенно включая более дальние. Такой подход обеспечивает более детальное изучение окружающей среды и улуч-

шает способность робота точно определить безопасные направления для движения, особенно в сложных и динамичных ситуациях. Если в ходе работы цикла оценки доступных секторов наступает момент, когда нет доступных направлений для движения – что может часто происходить в замкнутых пространствах с поворотами и ограничениями – робот возвращается к результату предыдущей итерации оценки. В этом случае он выбирает направление, которое было наиболее безопасным и близким к цели. Это позволяет роботу избежать застревания и продолжить движение к более открытому пространству, что особенно важно для обеспечения беспрепятственного передвижения в сложной обстановке. Если в процессе работы цикла расстояние до препятствия достигает или превышает дальность его обнаружения, равную 3,5999 единицы координатной сетки, то робот выбирает направление движения, которое совпадает с направлением к заранее заданной цели. Это позволяет ему максимально эффективно направляться к цели, несмотря на наличие либо отсутствие препятствий.

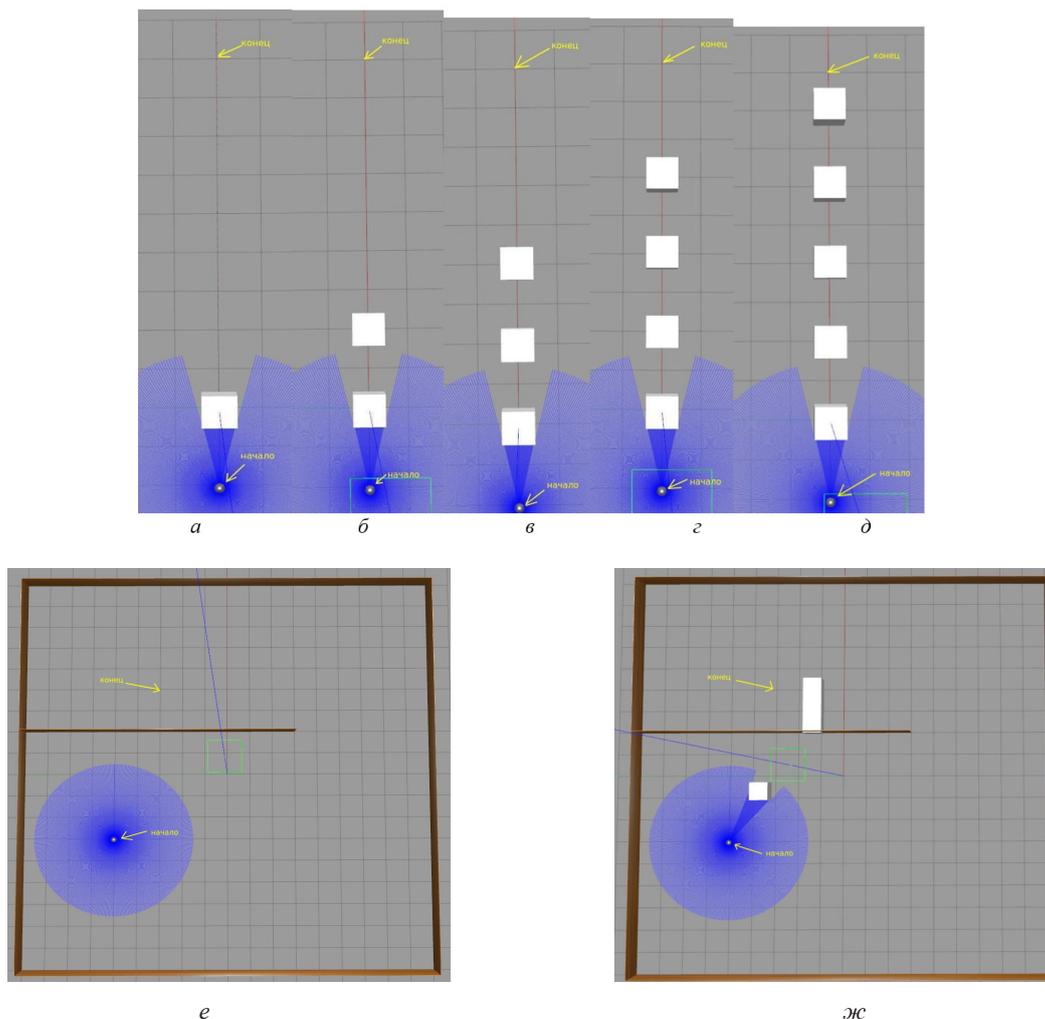


Рисунок 2. Симуляционное пространство в пакете визуализации Gazebo 11 с разными вариантами препятствий: кубического типа: *а* – одно препятствие, *б* – два препятствия, *в* – три препятствия, *г* – четыре препятствия, *д* – пять препятствий, а также: *е* – препятствие в виде длинной стены, *ж* – сложное препятствие

В алгоритме *vfh* роботы могут адаптировать свои действия в ответ на изменения в окружении. Когда робот движется вдоль препятствия и начинает удаляться от заданной цели, он перенастраивает своё направление движения. Вместо продолжения движения вдоль препятствия в том же направлении, робот выбирает изменить курс и начать двигаться вдоль препятствия в другую сторону. Модификация алгоритма *vfh* сосредоточена в том моменте, когда робот находится близко от препятствия и двигается вдоль него. В какой-то момент времени робот начинает удаляться от цели. Мы совмещаем алгоритм I с алгоритмом *vfh*, и, когда робот начинает отходить от цели, активируем алгоритм I, который заставляет робота продолжать движение выбранным направлением вдоль препятствия, пока он не приблизится к цели, обходя препятствие и двигаясь близко к нему. Когда робот сталкивается с препятствием (т.е. когда расстояние до него не превышает 0,88 единицы координатной сетки), он разворачивается в сторону препятствия, которая была первоначально выбрана, и начинает двигаться вдоль него по прямой. Это продолжается до тех пор, пока изгиб препятствия не позволит ему отдалиться от препятствия на 0,96 единицы координатной сетки. Затем робот поворачивается на 90 градусов в сторону препятствия и продолжает движение до тех пор, пока не отдалится на расстояние 0,96 единиц координатной сетки или не встретит новое препятствие (то есть не приблизится на расстояние менее 0,88 единиц координатной сетки). Если робот успешно отдалился на расстояние 0,96 единиц координатной сетки, он снова поворачивается на 90 градусов в том же направлении, в котором двигался ранее. Затем он продолжает движение в этом направлении, ожидая дальнейших препятствий или изменений в окружении. Как только робот зафиксирует приближение к цели при движении, он переключается обратно на алгоритм *vfh*. Это позволяет ему снова сосредоточиться на навигации к цели, учитывая окружающую обстановку и избегая препятствий на своем пути.

Анализ результатов вычислительного эксперимента

Результаты в виде временных характеристик (времени огибания) препятствий различного типа для алгоритмов I и II, приведены на рисунке 3. В процессе вычислительного эксперимента проводилась статистическая обработка полученных результатов. Для подтверждения эффективности предложенного алгоритма было проведено 10 экспериментов для каждого типа препятствий, по результатам которых рассчитывалось среднее время огибания каждого типа препятствий для алгоритмов I и II, погрешность определялась с использованием коэффициента Стьюдента с доверительной вероятностью $p = 0,95$.

При огибании одного кубического препятствия (рисунок 3) при использовании алгоритма II время оказалось в 2,5 раза меньше чем при использовании алгоритма I. Огибание двух кубических препятствий (рисунок 3) происходило в 3,7 раза быстрее чем в случае алгоритма I. Огибание трех кубических препятствий позволило ускорить процесс огибания в 5,4 раза относительно времени при использовании алгоритма I.

Среднее время огибания препятствия, минут

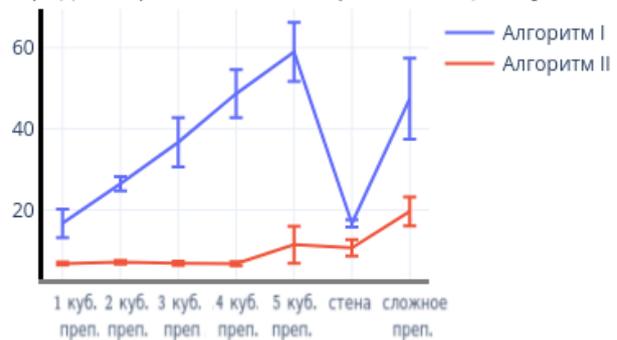


Рисунок 3. Графики зависимости оценки среднего времени огибания препятствий различного типа для алгоритма I и алгоритма II

При огибании групп из четырех и пяти кубических препятствий, длинной стены и сложного препятствия время огибания существенно снизилось при использовании алгоритма II: в 7,2 раза, в 5,2 раза, на 57 % и в 2,4 раза.

Заключение

В данной работе нами предложен алгоритм II огибания препятствий роботизированной системой, в котором отсутствует алгоритм машинного обучения. Проведена оценка времени движения до цели с огибанием препятствий в пространстве визуализации Gazebo 11.

Проведен вычислительный эксперимент по огибанию различного вида препятствий: кубического типа, а также препятствия в виде длинной стены, сложного препятствия с применением данного алгоритма. Выявлено, что с использованием алгоритма II движение робота до цели без столкновения с препятствием происходит быстрее, чем при использовании алгоритма I.

При использовании предложенного алгоритма II для каждого из рассмотренных в модели препятствий приводится пример движения робота в виде записанного видео. Видео размещено на сайте You Tube [5].

Верификация предложенного нами алгоритма II огибания препятствий при перемещении мобильного робота позволила установить преимущества по временным параметрам при сравнительном анализе с алгоритмом, приведенном в работе [1].

ЛИТЕРАТУРА

1. Сидоренко, А.В. Огибание препятствий при перемещении мобильного робота / А.В. Сидоренко, Н.А. Солодухо // Системный анализ и прикладная информатика. – 2023. – № 1. – С. 4-9.
2. Ulrich, I. VFH*: Local Obstacle Avoidance with Look-Ahead Verification / I. Ulrich, J. Borenstein // Proceedings of the 2000 IEEE International Conference on Robotics & Automation, San Francisco, CA, San Francisco, 2000. – Pp. 2505-2511.
3. Сидоренко, А.В. Алгоритм для огибания препятствий при движении мобильного робота / А.В. Сидоренко, Н.А. Солодухо // Компьютерные технологии и анализ данных (СТДА'2024) : материалы IV Междунар. науч.-практ. конф., Минск, 25-26 апр. 2024 г. / БГУ; редкол.: В. В. Скакун (гл. ред.), Н.Н. Яцков, В.В. Гринёв. – Минск : БГУ, 2024. – С. 122-125.
4. GitHub [Электронный ресурс]. – Режим доступа: <https://github.com/MikitaSaladukha/ROSproject> – Дата доступа: 11.03.2024.
5. YouTube [Электронный ресурс]. – Режим доступа: https://youtube.com/playlist?list=PLU4iF0_W9xOks7JPA_BAWjbfVECA8PgO4&si=GrTnC6w58-Rpeu8w – Дата доступа: 07.05.2024.

REFERENCES

1. Sidorenko A.V., Saladukha N.A. Bending obstacles when moving a mobile robot. System Analysis And Applied Information Science. 2023. № 1, pp. 4-9.
2. Ulrich I., Borenstein J. VFH*: Local Obstacle Avoidance with Look-Ahead Verification. Proceedings of the 2000 IEEE International Conference on Robotics & Automation, San Francisco, CA, San Francisco, 2000, pp. 2505-2511.
3. Sidorenko, A.V., Saladukha N.A. Algoritm dlya ogibaniya prepyatstvij pri dvizhenii mobil'nogo robota. Komp'yuternye tekhnologii i analiz dannyh (STDA'2024) : materialy IV Mezhdunar. nauch.-prakt. konf., Minsk, 25–26 apr. 2024. / Belarus. gos. un-t ; redkol.: V. V. Skakun (gl. red.), N.N. Yackov, V.V. Grinyov. – Minsk : BGU, 2024.– Pp. 122-125.
4. GitHub [Electronic resource]. Mode of access: <https://github.com/MikitaSaladukha/ROSproject>. Date of access: 11.03.2024.
5. YouTube [Electronic resource]. – Mode of access: https://youtube.com/playlist?list=PLU4iF0_W9xOks7JPA_BAWjbfVECA8PgO4&si=GrTnC6w58-Rpeu8w – Date of access: 07.05.2024

SIDORENKO A.V., SALADUKHA N.A.

OBSTACLE AVOIDANCE ALGORITHM IN MOBILE ROBOT MOTION

Belarusian State University
Minsk, Republic of Belarus

A method has been proposed, and a computer program has been developed for simulating obstacle avoidance by a robotic system. To expand the mobility capabilities of the robotic system, the algorithm combines the core elements of the vfh algorithm and our previously developed algorithm I [1], and is positioned in this work as algorithm II. An evaluation of the average obstacle avoidance time for various types of obstacles has been conducted, including five cubic obstacles, a long wall obstacle, and a complex obstacle. During the computational experiment using the Gazebo 11 simulation environment, the time parameters for robot movements around various types of obstacles were calculated, including five cubic obstacles, a wall obstacle, and a complex obstacle. In the computational experiment, statistical processing of the obtained results was carried out. It was shown that using the proposed algorithm, the assessment of the average time for the robotic system to avoid obstacles is reduced in some cases by up to 7.2 times compared to the use of algorithm I.

Keywords: robot, movement, obstacles, ROS 2, Gazebo 11



Сидоренко А.В., профессор факультета радиофизики и компьютерных технологий Белорусского государственного университета, доктор технических наук.

E-mail: sidorenkoa@yandex.by



Солодухо Н.А., старший преподаватель факультета радиофизики и компьютерных технологий Белорусского государственного университета, магистр физико-математических наук.