

Алгоритм планирования траектории мобильного робота

Нимаи Чандра Дас¹, А.Д. Скакун²
Санкт-Петербургский государственный
электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)
Санкт-Петербург, Россия

¹ nimaidey67@gmail.com, ² adstotskaya@etu.ru

Зиаул Хак Зим³, Райхан Уддин⁴
Международный университет Даффодил
Дакка, Бангладеш
³ ziaul15-1133@diu.edu.bd,
⁴ raihan35-1319@diu.edu.bd

Аннотация. В последние десятилетия разработки в области навигации беспилотных наземных транспортных средств (USV) и автономных мобильных роботов доказывают стремительное развитие данных технологий в области исследования помещений, замкнутых пространств и обхода препятствий. За счет устойчивости к небезопасным условиям, беспилотные наземные транспортные средства и автономные мобильные роботы имеют широкое применение в исследовании малоизвестных областей и быстром обнаружении опасных артефактов. Различные исследования были нацелены на усовершенствование приводных механизмов, планирование траектории движения и стратегий отслеживания траектории для роботов. Однако для небольших роботов, которые содержат чувствительные датчики и не оснащены защитными средствами, отсутствие мер по обходу препятствий снижает применимость таких устройств на практике. В данной работе рассматривается алгоритм обнаружения и предотвращения препятствий, разработанный для миниатюрных беспилотных наземных транспортных средств и автономных мобильных роботов. Детально обсуждаются процедуры построения траектории и определения относительного направления движения в условиях мультиагентной среды, а также процедуры оценки расстояния между роботами. В работе рассматривается решение одним алгоритмом двух задач: предотвращение столкновений с другим роботом или препятствием, измерение расстояния между главным агентами и другим агентом. Также в работе представлены экспериментальные результаты проверки алгоритма.

Ключевые слова: мобильный робот; планирование траектории; мультиагентная система; алгоритм

I. ВВЕДЕНИЕ

В настоящее время исследования, связанные с разработкой автономных мобильных роботов, являются приоритетной областью разработок в области искусственного интеллекта и робототехники. При этом спонтанное движение роботизированных агентов все еще является сложной задачей. При движении к цели агентам необходимо избегать других объектов в окружающей среде для предотвращения столкновений. Изучение и распознавание образов (агентов, робототехнических объектов, препятствий и т. д.) в комплексной системе зависит от наборов относительных ориентиров: прямого и обратного, левого и правого. Наиболее распространенной

задачей в настоящее время является реализация алгоритмической процедуры, когда одному агенту необходимо следить за относительными положениями других объектов мультиагентной робототехнической системы (МАРС). Следует отметить и дополнительную задачу, решение которой необходимо для реализации системы межагентного взаимодействия – предотвращение пересечения траекторий.

В настоящей работе продемонстрирован алгоритм управления роботом Path Finding Robot Avoiding Other Robot (PFRAOR), основанный на результатах [1]) для отыскания возможного и безопасного пути, который не только не допускает столкновений, но и точно изменяет расстояние до потенциально опасных объектов.

II. МУЛЬТИАГЕНТНАЯ СРЕДА

Мультиагентная среда (система, структура) – это современный способ управления, обеспечивающий интеллектуальное взаимодействие агентов внутри некоторого пространства. Мультиагентная среда может использоваться для решения задач, которые сложно или невозможно решить, используя традиционные алгоритмы управления. Мультиагентные структуры обычно содержат группу агентов и набор правил, которые обеспечивают взаимодействие между ними. Каждый агент выбирает правильное действие для решения задачи, используя несколько входных данных, например, историю действий, взаимодействий с соседними агентами и собственную цель.

Интеллектуальные агенты подразделяются на несколько типов в зависимости от их функций и механизмов принятия решений (см. рис. 1). Реактивные агенты принимают решения, используя только текущую информацию, не обращаясь к историческим данным. Таким образом, они используют прямое отображение ситуации в действие и напрямую реагируют на окружающую среду. Например, в методы реактивного агента используются для построения моделей слежения за автомобилем на основе искусственных нейронных сетей. Логические агенты принимают решения посредством логической дедукции. Известны методы обработки нескольких гипотез и выполнения диагностики неисправностей на основе логики.



Рис. 1. Характеристики среды мультиагентной системы

Интеллектуальные агенты «убеждение-желание-намерение» (belief, desire, and intention (BDI) построены с использованием символических представлений о намерениях, убеждениях и желаниях агентов. Обеспечение потенциальной автономии обеспечивается с использованием парадигмы BDI. Многоуровневые архитектуры включают несколько уровней программного обеспечения. Каждый уровень объединяет агентов, которые имеют дело с разными абстрактными уровнями среды.

III. ОБЗОР ЛИТЕРАТУРНЫХ ИСТОЧНИКОВ

В качестве обоснования актуальности и востребованности рассматриваемой в работе тематики, следует произвести обзор источников, посвященных последним результатам исследований в области интеллектуализации робототехнических систем и применяя методов искусственного интеллекта (ИИ) в задачах группового управления.

С целью решить проблемы традиционных методов планирования пути для мобильных роботов, в [1] используется фильтр Калмана для объединения данных и локализации, что позволяет мобильным роботам использовать этот метод для получения более высокой точности позиционирования. Чтобы лучше реализовать планирование пути мобильного робота, предлагается улучшенный метод динамического потенциального поля (Dynamic Artificial Potential Field Method) для преодоления недостатков традиционного подхода. Для проверки предложенного алгоритма планирования пути гусеничного мобильного робота в статической сложной среде тестируется путем моделирования в лабораторных условиях. Результаты моделирования показывают, что улучшенный алгоритм планирования пути позволяет мобильному роботу плавно обходить препятствия в сложном и узком пространстве и реализовывать оптимальное планирование пути для достижения поставленной цели.

В [2] предложен алгоритм поддержания траектории, который использует принцип уклонения, основанный на

моделировании поведения нескольких роботов с помощью алгоритма всенаправленной визуальной одновременной локализации и отображения (omni-directional visual simultaneous localization and mapping, OVSLAM) на основе изображений, полученных с камеры. OVSLAM извлекает характерные точки с помощью алгоритма Лукаса–Канаде (Lukas-Kanade OpticalFlow, LKOF) из изображений камеры и создает локальную карту, связывая координаты внешних точек. Он создает глобальную карту на основе информации о векторе и положении объектов, полученной из локальной карты, для определения метода избегания в соответствии с типом объекта, который классифицируется путем анализа одометрии, векторной информации и информации о положении. Были проведены два эксперимента, чтобы оценить производительность предложенного алгоритма. На основе результатов эксперимента с использованием робота, оснащенного реальным алгоритмом в различных средах, подтверждена эффективность распознавания и избегания препятствий, а также поддержания траектории.

В [3] предлагается улучшенный алгоритм A* (расширение алгоритма Дейкстры), основанный на введении родительского узла и изменении веса функции эвристической стоимости достижения рассматриваемой точки для управления планированием пути интеллектуальных роботов. Исследования моделированием подтвердили эффективность алгоритма как для модельной (имитационной), так и для реальной среды. Результаты моделирования показывают, что система с улучшенным алгоритмом A*, является устойчивой, время создания пути сокращается. Также отмечается, что производительность системы значительно выше, чем при использовании обычного алгоритма A*.

В [4] отмечается, что за последнее десятилетие было предложено множество подходов для решения множества задач планирования пути робота. Алгоритм оптимизации роя частиц (particle swarm optimization, PSO) - один из успешных методов оптимизации в этой области. В работе представлено успешное улучшение алгоритма PSO – D2PSO – который устраняет некоторые недостатки стандартного метода и не нарушает характеристик быстрой сходимости PSO, поскольку не влияет на базовую концепцию. Экспериментальные результаты показывают, что подход лучше работает при выходе из локального оптимума и доказывает, что применение D2PSO к задаче планирования пути для нескольких роботов является практичным и эффективным для большого количества роботов в средах с динамическими препятствиями.

С момента создания [5, 6] в научном сообществе обсуждается различные подходы к усовершенствованию PSO с точки зрения производительности. Известно, что алгоритм PSO имеет приемлемую производительность в поисковой задаче [7–9]. В [10] рассматривается вопрос планирования движения нескольких роботов с помощью дифференциальной эволюции (Differential Evolution, DE) и решается проблема как централизованным, так и распределенным подходами. Результаты оценки эффективности подтверждают преимущество второго подхода перед первым. Предлагаемый распределенный

алгоритм отличается от классических алгоритмов двухфазного планирования движения. Также было обнаружено, что распределенный алгоритм на основе DE сравним или превосходит алгоритм планирования пути на основе PSO.

IV. РАЗРАБОТКА АЛГОРИТМА

A. Вводные замечания

Для разработки алгоритма управления автономным мобильным роботом, в настоящем исследовании используется методология, представленная теорией алгоритма PFRAOR. Прежде чем формализовать модель, необходимо исследования указать числовые характеристики алгоритма PFRAOR и уточнить свойства, которые представляют собой основу для иллюстрации основной гипотезы. Программная структура алгоритма может быть описана следующим образом:

Случай-1: главный агент движется вперед, а второй агент движется вправо.

Случай-2: главный агент движется вперед, а второй агент движется влево.

Случай-3: главный агент движется вперед, а второй агент движется навстречу.

Случай-4: главный агент движется вперед, а второй агент также движется вперед.

Случай 5: главный агент движется вперед, второй агент приближается справа, третий агент приближается слева.

Случай 6: главный агент движется вперед, второй агент движется навстречу к главному агенту, третий агент приближается слева.

Случай-7: главный агент движется вперед, второй агент движется навстречу к главному агенту, третий агент приближается справа.

B. Структура алгоритма

Разработанный алгоритм PFRAOR расчета дистанции от главного агента до других участников взаимодействия, обеспечивает поддержание определенного расстояния и достижения поставленной цели движения. Алгоритм имеет программную реализацию, кратко описать которую можно следующим образом:

Шаг 1. Создание агента с указанием направления.

- Массив *Направление* = [right, left, forward, backward]. Переменные точки направления главного агента – r, l, f и b (вправо = r, влево = l, вперед = f и назад = b).
- Массив *Направление движения* = [ToRight, ToLeft, ToForward, ToBackward]. Переменными точки направления движения другого агента являются: tr, tl, tf и tb. Точки направления содержат разные значения: TurnRight, TurnLeft, TurnForward и TurnBack.

Шаг 2. Реализация метода проверки движения агента.

Шаг 3: Реализация метода определения места назначения (цели) – проверить новое движение главного агента; инициализировать новое место назначения.

Шаг 4: Создание цикла

Шаг 5: Определение места назначения

- если *prevDestination* = 'f' и *mainAgent* = 'r',
 to *destination* = 'l';
 ...
 иначе, если *prevDestination* == 'l' и *mainAgent* == 'f',
 to *destination* = 'l';
 mainAgent = пункт назначения;
- Переход к шагу 2

Шаг 6: Завершить выполнение алгоритма.

C. Алгоритм измерения расстояния

Для проведения экспериментов использовался ультразвуковой датчик HC-SR04 для измерения расстояния между двумя роботизированными агентами, который способен измерять расстояние от 2 до 450 см. Последовательность действий алгоритма может быть представлена как: инициализация контакт датчика, посылка сигнала к препятствию, получение сигнала, расчет расстояния, конец

V. МЕТОДЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В РЕШАЕМОЙ ЗАДАЧЕ

Искусственный интеллект - это область разработки программного обеспечения и программирования, которая позволяет вычислительным средствам расширять набор функций путем обучения. Методы ИИ в том числе включают исследование и разработку вычислительных алгоритмов, которые могут обрабатывать информацию и формировать выводы и рекомендации, делая выбор на основе прогностических моделей.

В данном разделе описывается метод, основанный на относительном расчете данных, основанном на информации нескольких агентов. Важно отметить, что практические источники информации в будущем могут быть лучше проработаны методами ИИ.

A. Методика проведения эксперимента

Целью экспериментального (количественного) исследования является проверка применимости разработанных числовых моделей и алгоритмов. Изложенная ниже схема количественной оценки была разработана в рамках алгоритмического исследования. Также следует отметить, что на результат влияют и вычислительные.

Объектом исследования является робот, следующий по рассчитанной (в зависимости от цели) траектории и избегающий столкновения с другими роботами (согласно алгоритму, PFRAOR).

В. Исследование точности

Окончательное движение главного агента относительно других агентов определяется как: T_r = поворот направо, T_f = поворот вперед, T_l = поворот налево, T_{ff} = поворот вперед, T_{tf} = поворот вперед. W = ждать. Результаты эксперимента приведены в табл. 1.

ТАБЛИЦА I Окончательное перемещение главного агента

Случай	Направление		Заключит. движение
	Другие агенты	Главный агент	
1	tr	tf	tl
2	tl	tf	tr
3	tff	tff	tr
4	tff	tff	tl
5	tf	ttf	tf
6	tr	tl	tl
7	tr	tff	tr
2	tl	tf	tr

С. Инициализация пункта назначения

Для инициализации используются следующие обозначения: R = правый, F = вперед, L = слева, B = назад. Результат инициализации представлен в табл. 2.

ТАБЛИЦА II Инициализация пункта назначения

Заключительное движение	Предыдущий пункт назначения	Пункт назначения
r	f	l
r	r	f
r	b	r
r	l	b
b	f	f
b	r	r
b	b	b
b	l	l
l	f	r
l	r	b
l	b	l
l	l	f
f	f	f
f	r	r
f	b	b
f	l	l

VI. СРАВНЕНИЕ С ИЗВЕСТНЫМ РЕЗУЛЬТАТОМ

Рассмотрим сравнительный анализ результатов функционирования, алгоритма PFRAOR, полученного в данной работе, и известного из [2] алгоритма OVSLAM. Было произведено сравнительное исследование моделированием и доказано, что алгоритм PERAOR обеспечивает перемещение главного агента в целевое местоположение при одновременном наблюдении за другими агентами. Результаты сравнительного исследования представлены в табл. 3.

ТАБЛИЦА III СРАВНЕНИЕ РАБОТЫ АЛГОРИТМОВ PFRAOR И OVSLAM

Алгоритм	Предотвращение столкновений	Измерение расстояния
PFRAOR	ДА	ДА
OVSLAM	ДА	НЕТ

VII. ЗАКЛЮЧЕНИЕ

Представлен комбинированный алгоритм Path Finding Robot Avoiding Other Robot (PFRAOR) управления автономным мобильным роботом, обеспечивающий расчет траектории на основе принятой цели, движение вдоль принятой траектории без столкновения с другими объектами мультиагентной системы или препятствиями, расчет расстояния до потенциально опасных объектов.

Проведенное сравнение с известным алгоритмом – всенаправленной визуальной одновременной локализации и отображения, OVSLAM – доказывает преимущество предложенного алгоритма PFRAOR за счет возможности точного определения расстояния, информация о котором может быть использована для решения смежных задач.

СПИСОК ЛИТЕРАТУРЫ

- [1] Xu R. Path planning of mobile robot based on multi-sensor information fusion // EURASIP Journal on Wireless Communications and Networking. 2019. Т. 2019. №. 1. С. 1-8.
- [2] Choi Y.W. et al. Multi-robot avoidance control based on omnidirectional visual SLAM with a fisheye lens camera // International Journal of Precision Engineering and Manufacturing. 2018. Т. 19. №. 10. С. 1467-1476
- [3] Lin M. et al. Path planning of mobile robot based on improved A* algorithm // 2017 29th Chinese Control And Decision Conference (CCDC). IEEE, 2017. С. 3570-3576.
- [4] Ayari A., Bouamama S. A new multiple robot path planning algorithm: dynamic distributed particle swarm optimization // Robotics and biomimetics. 2017. Т. 4. №. 1. С. 1-15.
- [5] Kennedy J., Eberhart R. Particle swarm optimization // Proceedings of ICNN'95-international conference on neural networks. IEEE, 1995. Т. 4. С. 1942-1948.
- [6] Eberhart R., Kennedy J. A new optimizer using particle swarm theory // MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science. IEEE, 1995. С. 39-43.
- [7] Doctor S., Venayagamoorthy G.K., Gudise V.G. Optimal PSO for collective robotic search applications // Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753). IEEE, 2004. Т. 2. С. 1390-1395.
- [8] Pugh J., Segapelli L., Martinoli A. Applying aspects of multi-robot search to particle swarm optimization // International Workshop on Ant Colony Optimization and Swarm Intelligence. Springer, Berlin, Heidelberg, 2006. С. 506-507.
- [9] Hereford J.M. A distributed particle swarm optimization algorithm for swarm robotic applications // 2006 IEEE International Conference on Evolutionary Computation. IEEE, 2006. С. 1678-1685.
- [10] Chakraborty J. et al. Distributed cooperative multi-robot path planning using differential evolution // 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence). IEEE, 2008. С. 718-725.