

Программная система визуализации маршрутов движения роботов по оптимальным траекториям внутри промышленного цеха

Т.М. Кузьмина, О.А. Ветрова, А.А. Белевитин

Российский государственный университет им. А.Н. Косыгина (Технологии. Дизайн. Искусство), Москва

Аннотация: В автоматизации производства важным направлением является использование роботов. При разработке алгоритмов управления перемещением такой машины возникают разнонаправленные задачи. В критической ситуации робот не должен нанести вред работникам или столкнуться со своим собратом. При этом машина должна двигаться по оптимальному маршруту. В статье проведен анализ методов поиска оптимальных путей на квадратных навигационных сетках. Для исследования и проверки алгоритмов разработана система визуализации и тестирования на языке C#. Уникальной функцией системы является возможность непосредственного считывания изображений карт цехов в широко распространенных графических форматах и автоматическое преобразование их в формат, понятный системе. Анализ полученных результатов позволяет сделать вывод о том, что лучше использовать метод, который допускает непостоянную или неизвестную заранее структуру сетки.

Ключевые слова: визуализация, программа, графический формат, робот, цех, карта, квадрат, сетка, ячейка, навигация, оптимальный маршрут, планарный граф.

В связи с тем, что роботизация является важным направлением в автоматизации производства, рассмотрим особенности движения робота внутри промышленного цеха. При разработке алгоритмов управления перемещением такой машины возникают разнонаправленные задачи, в которых исследуются и выделяются свои характерные особенности [1, 2]. Например, в работе [1] внимание акцентируется на условиях неопределенности и заранее неизвестных математических моделях движения мобильных роботов. А в работе [2] важной особенностью поиска оптимальных траекторий движения является учет дополнительных работ, сопровождающих движение объекта.

Промышленные роботы обычно выполняют однотипные операции и прикрепляются к одному рабочему месту рядом с конвейерной линией. Каждая такая операция имеет строгий алгоритм, который программируется. При программировании движения робота внутри помещения нужно

учитывать внезапно возникающие препятствия. В критической ситуации такая машина не должна нанести вред работникам или столкнуться со своим собратом роботом. Поэтому, прежде чем программировать движения роботов внутри цеха, необходимо исследовать существующие методы поиска оптимальных маршрутов, и выбрать лучший, применительно к конкретному цеху.

Начнем с представления рабочего пространства. Квадратная навигационная сетка соответствует неориентированному планарному графу [3]. Пути движения объекта в таком графе возможны в любом направлении окрестности фон Неймана [4, 5].

В окрестности выделяют три типа квадратных ячеек. По ячейкам первого типа (без препятствий) объект может свободно двигаться. Ячейки первого типа окрашены в белый цвет. Ячейки второго типа обозначают любые постоянные препятствия: стены, оборудование и так далее. Ячейки второго типа окрашены в красный цвет. Ячейки третьего типа (с переменными помехами) отвечают за возможное появление человека или аналогичного робота по пути следования. Ячейки третьего типа окрашены в желтый цвет. Каждая ячейка есть квадрат со стороной в одну масштабную единицу.

Сравним существующие методы поиска оптимальных путей на неориентированных планарных графах.

Волновой метод основан на поиске в ширину и состоит из двух этапов: распространение волны и восстановление пути [6, 7]. В начале первого этапа из стартового элемента распространяется волна в четырех направлениях. Затем выявляют ячейки, в которые пришла волна. Эти ячейки образуют элементы первого фронта волны. Каждый элемент первого фронта волны является источником вторичного фронта волны. Элементы второго фронта генерируют волну третьего фронта и т.д. Процесс продолжается до тех пор,

пока не будет достигнут конечный фронт или сделан вывод о невозможности построения пути. Восстановление пути происходит от элементов конечного фронта к стартовому элементу [6, 7].

К основным недостаткам волнового метода можно отнести большой объем необходимой памяти при реализации на ЭВМ; отсутствие приоритетных направлений поиска, так как распространение волны происходит во все стороны одинаково; необходимость в новом перерасчете всего дискретного поля сетки после любого изменения в рабочем пространстве графа [6, 7].

Алгоритм A^* представляет собой расширение алгоритма Дейкстры [8]. Для неориентированного планарного графа алгоритм A^* базируется на методе построения оптимального пути по первому лучшему совпадению [9, с. 70-80]. Эвристическая функция определяет порядок обхода вершин. Некорректный баланс этой функции может привести к неоптимальным результатам и прохождению ненужных областей. Функция должна быть сбалансирована по минимальному расстоянию, которое надо пройти от начала пути к текущей вершине [10]. Для балансировки можно использовать Манхэттенское расстояние из-за ограничений, накладываемых перемещением робота по окрестностям фон Неймана [4, с. 44-46, с. 62-63].

Алгоритм A^* имеет ряд преимуществ перед волновым методом. Он обеспечивает меньшее количество пройденных вершин при правильно подобранной эвристической функции. При этом посещенные вершины не ведут к конечной точке, что приводит к сокращению времени поиска оптимального маршрута, но сохраняется недостаток в необходимости постоянного перерасчета при любых изменениях графа.

Алгоритм D^* Lite применяют для поиска кратчайшего пути во взвешенном ориентированном графе, где структура графа неизвестна заранее или постоянно подвергается изменению [11]. У нас рабочее пространство

перемещений робота представлено в виде неориентированного планарного графа, но, несмотря на это, можно попробовать в нашем случае применить алгоритм D* Lite для устранения необходимости перерасчета навигационной сетки.

Также в литературных источниках можно найти описания методов инкрементного эвристического поиска на графах [12, 13]. Эти методы для сокращения времени решения поисковых задач используют эвристику для фокусировки поиска и повторного использования информации из предыдущих поисков.

Для исследования и проверки существующих методов была разработана система визуализации и тестирования на языке C# с использованием программной среды IDE Microsoft Visual Studio. Внешняя библиотека GraphX была выбрана как инструмент визуализации методов. Для исследования и проверки методов был использован макет швейного цеха с обозначением контуров оборудования и другого рода препятствий. Макет выполнен в масштабе 1:100 (рис. 1).

Уникальной функцией разработанной системы является возможность непосредственного считывания изображений карт цехов в форматах (JPG, GIF, PNG, BMP) и автоматическое преобразование их в формат, понятный системе. Для реализации данной функции была использована внешняя библиотека машинного зрения Emgu CV. Вначале пользователь загружает план цеха (чем лучше качество изображения, тем выше шанс получить карту, требующую минимум доработок). Для более легкой обработки изображение преобразуется в два цвета: черный и белый. Потом изображение обрабатывается библиотекой Emgu CV, на выходе мы получаем все контуры неподвижных препятствий.

Далее в зависимости от размера робота накладывается ортогональная навигационная сетка. Для примера на рис. 1 представлен обработанный и разлинованный план цеха.

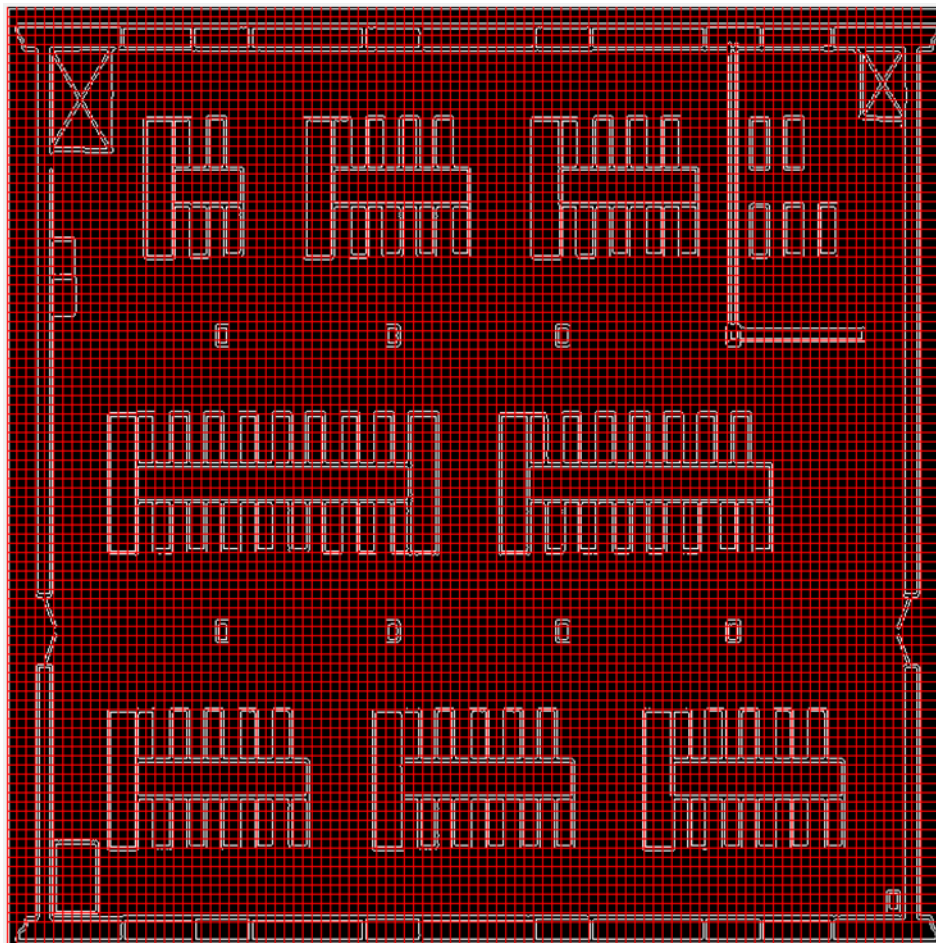


Рис. 1. – Схема швейного цеха с навигационной сеткой.

На рис. 2 показано, как эта схема будет выглядеть в программе тестирования.

В некоторых случаях карта переносится не точно, для редактирования карты можно воспользоваться инструментами, которые есть в программе тестирования. В ней же можно построить схему с «нуля» или откорректировать любую другую.

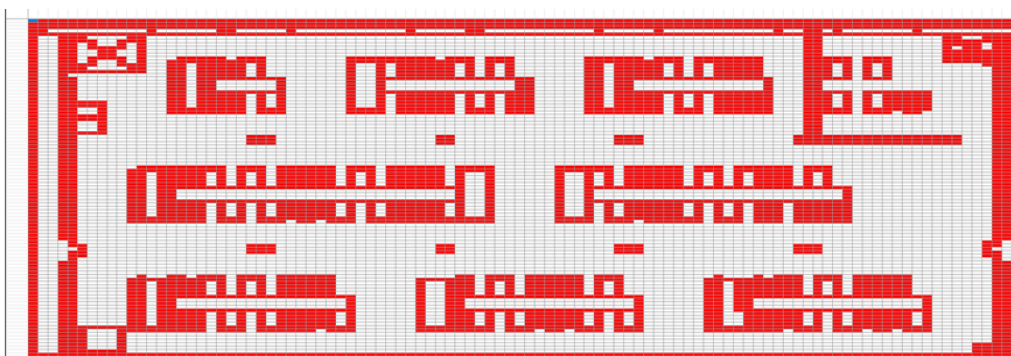


Рис. 2. – Карта цеха в программе тестирования

Поскольку помехи могут возникать не на всем пространстве движения, то на схеме отмечаются области, где их появление возможно. На рис. 3 показана схема, полностью подготовленная для испытаний. Желтым цветом закрашены критические области, в которых можно ожидать появления помех.

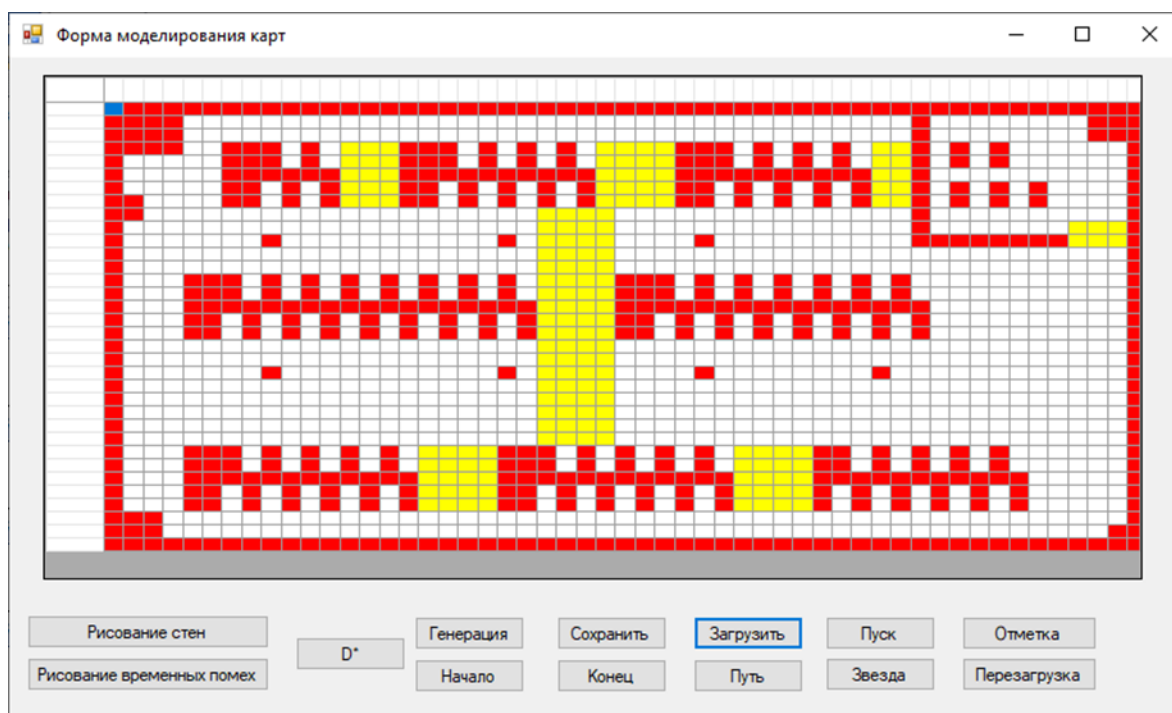


Рис. 3. – Схема цеха с отмеченными критическими областями.

С данной схемой были проведены испытания. Каждый метод запускался 100 раз с вероятностью 50 процентов появления препятствия для каждой клетки критической области. Начальная и конечная точки не менялись. Результаты представлены в таблице № 1.

Таблица № 1

Итоги испытаний методов

Параметр	Волновой метод	A*	D* Lite
Время выполнения (мс)	47.8	19.2	14.1
Шаги	70	72	65

Исходя из полученных результатов, можно сделать вывод о необходимости использования именно алгоритма D* Lite. Данный алгоритм показывает более быструю работу и меньшее количество шагов, пройденных роботом.

В заключение отметим, что данная статья развивает идеи, изложенные в работах [14, 15].

Литература

1. Гайдук А.Р., Капустян С.Г., Шаповалов И.О. Алгоритм управления движением группы мобильных роботов в условиях неопределенности // Инженерный вестник Дона, 2018, №3. URL: ivdon.ru/ru/magazine/archive/n3y2018/5221/.

2. Нечитайло Н.М., Панасов В.Л., Линденбаум Т.М. Оптимизация назначения средств доставки в транспортных задачах по критерию времени // Инженерный вестник Дона, 2021, №2. URL: ivdon.ru/ru/magazine/archive/n2y2021/6840/.

3. Алексеев В.Б. Граф плоский. Математическая Энциклопедия. Ред. коллегия: И.М. Виноградов (глав. ред.) [и др.] Т.1, М.: Издательство «Советская энциклопедия», 1977, с. 1109-1110.

4. Тоффоли Т., Марголюс Н. Машины клеточных автоматов. М.: Мир, 1991. 280 с.
 5. фон Нейман Дж. Теория самовоспроизводящихся автоматов. М.: Мир, 1971. 384 с.
 6. Moore E.F. The shortest path through a maze // Proceedings of an International Symposium on the Theory of Switching (Cambridge, Massachusetts, 2-5 April, 1957), Harvard University Press, 1959, Vol. 2, pp. 285-292.
 7. Lee C.Y. An Algorithm for Path Connections and Its Applications // IRE Transactions on Electronic Computers, 1961, vol. EC-10, number 2, pp. 364-365.
 8. Dijkstra E.W. A note on two problems in connexion with graphs. // Numerische Mathematic: Springer Science+Business Media, 1959, Vol. 1, Iss. 1, pp. 269-271.
 9. Нильсон Н. Искусственный интеллект: методы поиска решений. М.: Мир, 1973. 273 с.
 10. Hart P.E., Nilsson N.J., Raphael B. A Formal Basis for the Heuristic Determination of Minimum Cost Paths // IEEE Transactions on Systems Science and Cybernetics SSC4, 1968, № 2, pp. 100-107.
 11. Koenig S., Likhachev M. D*Lite // AAAI /IAAI, 2004, V. 25, № 2, pp. 99-112.
 12. Edelkamp S., Schrödl S. Heuristic search: theory and applications, Morgan Kaufmann Publishers, 2012, № 2, ISBN 978-0-12-372512-7.
 13. Рассел С., Норвиг П. Составление допустимых эвристических функций. Искусственный интеллект: современный подход. 2-е изд., 2006, М.: Вильямс, 1973, с. 170-173, ISBN 5-8459-0887-6.
 14. Кузьмина Т.М., Ветрова О.А. Использование компьютерной программы «алгоритмы на графах» в учебном процессе // Дизайн и технологии, 2019, №70(112), М.: РГУ им. А.Н. Косыгина, с. 135-139.
-

15. Кузьмина Т.М., Ветрова О.А. Использование linq-запросов в многокритериальном поиске кратчайшего пути при лексикографическом упорядочивании критериев // Сборник научных трудов Международного научно-технического симпозиума «Современные инженерные проблемы в производстве товаров народного потребления» Международного Косыгинского форума «Современные задачи инженерных наук», 2019, Часть 3, М.: РГУ им. А.Н. Косыгина, с. 82-87

References

1. Gajduk A.R., Kapustjan S.G., Shapovalov I.O. Inzhenernyj vestnik Dona, 2018, №3. URL: ivdon.ru/ru/magazine/archive/n3y2018/5221/.
2. Nechitajlo N.M., Panasov V.L., Lindenbaum T.M. Inzhenernyj vestnik Dona, 2021, №2. URL: ivdon.ru/ru/magazine/archive/n2y2021/6840/.
3. Alekseev V.B. Graf ploskij [Planar graph]. Matematicheskaja Jenciklopedija [Mathematical Encyclopedia]. Red. kolegija: I.M. Vinogradov (glav. red.) [i dr.] T.1, М.: Izdatel'stvo «Sovetskaja jenciklopedija», 1977, pp. 1109-1110.
4. Toffoli T., Margolus N. Mashiny kletochnyh avtomatov [Cellular Automata Machines]. М.: Mir, 1991. 280 p.
5. фон Nejman Dzh. Teorija samovosproizvodjashhihsja avtomatov [Theory of Self-Reproducing Automata]. М.: Mir, 1971. 384 p.
6. Moore E.F. Proceedings of an International Symposium on the Theory of Switching (Cambridge, Massachusetts, 2-5 April, 1957), Harvard University Press, 1959, Vol. 2, pp. 285-292.
7. Lee C.Y. IRE Transactions on Electronic Computers, 1961, vol. EC-10, number 2, pp. 364-365.
8. Dijkstra E.W. Numerische Mathematic: Springer Science+Business Media, 1959, Vol. 1, Iss. 1, pp. 269-271.
9. Nil'son N. Iskusstvennyj intellekt: metody poiska reshenij [Problem-Solving Methods in Artificial Intelligence]. М.: Mir, 1973. 273 p.



10. Hart P.E., Nilsson N.J., Raphael B. IEEE Transactions on Systems Science and Cybernetics SSC4, 1968, № 2, pp. 100-107.
11. Koenig S., Likhachev M. AAAI /IAAI, 2004, V. 25, № 2, pp. 99-112.
12. Edelkamp S., Schrödl S. Heuristic search: theory and applications. Morgan Kaufmann Publishers, 2012, № 2, ISBN 978-0-12-372512-7.
13. Rassel S., Norvig P. Sostavlenie dopustimyh jevristicalicheskikh funktsij [Drawing up valid heuristic functions]. Iskusstvennyj intellekt: sovremennyj podhod [Artificial Intelligence: A Modern Approach]. 2-e izd., 2006, M.: Vil'jams, 1973, pp. 170-173, ISBN 5-8459-0887-6.
14. Kuz'mina T.M., Vetrova O.A. Dizajn i tehnologii, 2019, №70(112), M.: RGU im. A.N. Kosygina, pp. 135-139.
15. Kuz'mina T.M., Vetrova O.A. Sbornik nauchnyh trudov Mezhdunarodnogo nauchno-tehnicheskogo simpoziuma «Sovremennye inzhenernye problemy v proizvodstve tovarov narodnogo potreblenija» Mezhdunarodnogo Kosygin'skogo foruma «Sovremennye zadachi inzhenernyh nauk», 2019, Chast' 3, M.: RGU im. A.N. Kosygina, pp. 82-87.