

Методы поиска допустимых путей для беспилотных летательных аппаратов

12, декабрь 2014

Ткачев С. Б., Виноградова М. С.

УДК 004.9+519.1

Россия, МГТУ им. Н.Э. Баумана

mathmod@bmstu.ru

Введение

Для беспилотных летательных аппаратов (БПЛА) различных конструкций [1] одной из активно исследуемых задач является задача обеспечения автономного функционирования. Для решения этой задачи необходимо на борту БПЛА осуществить автоматическое планирование маршрута движения и обеспечить реализацию выбранного маршрута. Например [4], один из типовых сценариев использования БПЛА предполагает посещение заранее сформированного списка контрольных точек, заданных своими координатами в пространстве, и выполнение в каждой точке определенного задания.

Если требуется найти пространственную кривую, проходящую через заданную последовательность путевых точек и удовлетворяющую дополнительным пространственным ограничениям, то говорят о прокладке допустимого пути. В качестве пространственных ограничений рассматривают, например, различные неподвижные препятствия. Закон изменения скорости при движении вдоль заданного пути задают отдельно.

Известны подходы к планированию допустимых путей движения БПЛА, основанные на анализе двумерной карты, на которую нанесены проекции препятствий [3]. Результатом работы алгоритмов планирования в этом случае является плоская кривая, которая рассматривается как проекция допустимого пути. Сам путь как правило проходит на заданной высоте и также задается плоской кривой.

Рассмотрение задач поиска допустимых путей на плоскости связано с тем, что задачи поиска допустимых путей в пространстве имеют существенно более высокую вычислительную сложность.

Далее представлен обзор методов, которые применяются для решения задачи поиска допустимых путей. Обсуждается возможность применения рассмотренных алгоритмов при решении задачи поиска в трехмерной среде.

В настоящее время известно достаточно много различных методов поиска путей в среде с препятствиями. В рамках обзора рассмотрены следующие группы методов:

- методы потенциалов и штрафных функций;
- методы, основанные на использовании генетических алгоритмов;
- методы поиска путей во взвешенных графах;

1. Метод потенциалов

Метод потенциалов основан на задании специальных функций, «запрещающих» прохождение путей движения объекта вблизи препятствия. Можно провести аналогию с потенциалом поля точечного электрического заряда. При приближении одноименного заряда к препятствию, в окрестности которого задано такое электрическое поле, на последний действует сила отталкивания, которая растет по мере приближения заряда к препятствию. Возможны и другие способы задания потенциалов, не обязательно основанные на физических аналогиях. В результате формируется достаточно сложная многоэкстремальная функция, формирующая «ландшафт», по которому должен проходить допустимый путь. Для организации обхода препятствий могут также задаваться различные функции штрафа.

Методы и алгоритмы, основанные на потенциалах, привлекают значительный интерес [5], однако имеют скорее теоретическое значение. К достоинствам соответствующих алгоритмов можно отнести достаточную гладкость кривой, получающейся в результате работы алгоритма. Однако результат работы существенно зависит от того, как заданы функции, определяющие потенциалы. При неудачном выборе этих функций решение задачи может быть не найдено, поскольку путь, прокладываемый из начальной точки в направлении убывания потенциала, может привести в локальную потенциальную яму, где алгоритм поиска остановится. Кроме того, при наличии сложного «ландшафта» для вычисления функции потенциала (или ее градиента) требуются существенные вычислительные ресурсы.

Хотя для преодоления проблемы локального минимума предложено значительное количество алгоритмов (см., в частности, [6, 7]), степень их универсальности неясна.

Известен, например, подход к решению этой проблемы, основанный на муравьином алгоритме [8]. В этом случае глобальный поиск осуществляется с использованием муравьиного алгоритма, а локальный — с использованием метода потенциальных полей. Другой подход предполагает использование метода потенциалов для глобального поиска. При остановке в точке локального минимума поиск начинается заново, причем точка локального минимума становится источником запрещающего потенциала. Однако даже улучшенный алгоритм не в состоянии обеспечить нахождение пути при достаточно сложных конфигурациях препятствий.

Для ограниченных классов препятствий найдены функции потенциала, при использовании которых не возникает проблема локального минимума [9], но такое решение требует существенных вычислительных затрат и также не является универсальным.

Успешным примером использования метода потенциалов в задаче планирования путей является решение задачи избегания столкновений с другими агентами в мультиагентных средах [10].

Следует отметить, что в для задания потенциалов используются различные физические аналогии из электростатики, динамики жидкости, газовой динамики и других разделов механики сплошных сред [11, 7]. Использование сложных аналогий, впрочем, существенно увеличивает вычислительную сложность задачи поиск пути. Например, для прокладки пути требуется решение задачи теплопроводности с использованием разностного или конечно-элементного подходов.

Отметим, что для решения трехмерных задач применение метода потенциалов теоретически возможно, однако при этом, по-видимому, придется отказаться от стандартного подхода, основанного на использовании методов оптимизации, поскольку путь, получаемый на основе таких методов, почти всегда будет огибать препятствия, в то время как более рациональным может оказаться путь, проходящий над препятствиями.

2. Генетические алгоритмы

Генетические алгоритмы, как правило, используются для нахождения набора опорных точек, формирующих допустимый путь. Найденные точки служат исходными данными для построения пути с использованием кривых Безье или В-сплайнов. Описание возможных реализаций можно найти в работах [12, 13]. По заданным опорным точкам ищется кривая в виде соответствующего сплайна, реализующая допустимый путь.

При этом в понятие «ландшафт» включены все возможные препятствия. Поскольку гладкую кривую довольно высокой сложности можно достаточно точно аппроксимировать В-сплайном с небольшим числом опорных точек, оказывается возможным обрабатывать большие популяции при умеренном использовании вычислительных ресурсов и находить достаточно хорошие решения. Другими достоинствами указанных методов являются возможность учета некоторых дополнительных ограничений на прокладываемый путь, а также возможность гибкого управления параметрами алгоритма.

К недостаткам можно отнести неполноту алгоритма (нет гарантии, что приемлемый результат вообще будет получен), отсутствие конкретных рекомендаций по настройке параметров алгоритма. В частности, не известно число опорных точек, необходимое для прокладки допустимого пути.

3. Поиск на графах

Возможно, наиболее общим и часто используемым подходом к поиску путей является использование графов, отражающих топологию исследуемой области. Можно выделить алгоритмы, предполагающие разбиение пространства на регулярные ячейки: правильные

симплексы, кубы или (в двумерном случае) правильные шестиугольники, а также алгоритмы, работающие с областями неправильной формы. Эти области, как правило, получаются в результате анализа расположенных в пространстве препятствий.

Каждая полученная ячейка ассоциируется с вершиной графа, вершины соединяются ребрами (или дугами) в соответствии с отношением непосредственной достижимости, заданным на множестве ячеек. Как правило, рассматриваются неориентированные графы, поскольку отношение непосредственной достижимости считается симметричным.

Каждому ребру ставится в соответствие некоторый вес. Для определения веса ребра графа обычно используется какая-либо метрика, заданная на множестве ячеек, хотя возможны и другие подходы, поскольку для корректной работы многих алгоритмов поиска не требуется, чтобы веса ребер удовлетворяли неравенству треугольника.

Для поиска пути в построенном взвешенном графе применяется один из алгоритмов поиска на графах. Выделяют полные и неполные методы поиска оптимального пути. Первые обеспечивают нахождение результата, если он существует. Если начальная и конечная вершины находятся в разных компонентах связности, полный алгоритм сообщит, что искомым путь не существует. Несмотря на то, что наличие свойства полноты выглядит желательным, в задачах с достаточно большими графами полный поиск может потребовать значительных вычислительных ресурсов и времени.

К полным алгоритмам поиска относится алгоритм Дейкстры, первая версия которого была представлена в 1959 г. [14]. Время работы алгоритма затем было улучшено за счет использования более подходящей структуры данных — очереди с приоритетом [15].

Алгоритм поиска в ширину [16] также может быть использован для нахождения минимального пути, однако при использовании классического варианта каждому ребру приписывается единичная длина. Такое ограничение может быть приемлемым в случае покрытия пространства кубическими ячейками (кубического замощения) с разрешенными переходами от текущей ячейки только к ячейкам, пересекающимся с ней по целой грани, в общем случае от этот алгоритм оказывается очень трудоемким.

Часто используемой разновидностью алгоритма Дейкстры и фактическим стандартом для поиска путей во многих приложениях является алгоритм A^* [17], представленный в 1968 г. Это пример алгоритма, производящего поиск по первому наилучшему совпадению, т.е. в первую очередь расширяющего поиск из наиболее перспективных узлов. Перспективность узла определяется эвристической функцией, оценивающей расстояние до цели. Если эвристика удовлетворяет определенным условиям, оптимальный путь будет найден быстрее, чем это можно сделать с использованием алгоритма Дейкстры. Собственно алгоритм Дейкстры является частным случаем алгоритма A^* при нулевой эвристике. Вычислительная сложность алгоритма A^* не превышает таковую для алгоритма Дейкстры и определяется согласованностью выбранной эвристики с реальной оценкой длины пути.

Перечисленные алгоритмы при практическом использовании могут потерпеть неудачу в случае анализа графа с очень большим количеством вершин и ребер. Это связано как

со временем работы, так и с тем обстоятельством, что эти алгоритмы предполагают хранение информации о пройденных вершинах. Поэтому занимаемый этой информацией объем памяти может оказаться неприемлемым.

Предложены модификации этих алгоритмов, работающие с ограниченным объемом памяти, в частности, SMA* (Simplified Memory Bounded A*) [18]. Такие алгоритмы, вообще говоря, не полны, т.е. могут не найти решение даже в случае его существования, зато используют заранее отведенный объем памяти.

Для ускорения поиска можно применять двунаправленные алгоритмы. Хотя алгоритмы с двунаправленным поиском вынуждены хранить гораздо больше информации, чем их однонаправленные аналоги, существует модификация с ограниченным использованием памяти [19], основанная на алгоритме SMA*.

Известна группа методов, использующих различные уровни детализации задачи поиска. Одним из таких является сравнительно новый (предложен в 2004 г.) алгоритм НРА* (Hierarchical Pathfinding A*) [20], а также его модификации ДНРА* и **, имеющие лучшее время работы [21]. Эти алгоритмы, однако, разрабатывались для поиска пути на плоских или многоуровневых картах. Для оценки возможности их применения в случае трехмерной среды требуется дополнительное исследование.

4. Подходы к учету ограничений и сглаживанию путей

Путь, полученный с использованием любых алгоритмов, должен удовлетворять определенным дополнительным ограничениям, обусловленным областью его применения [22]. Например, такими ограничениями могут быть ограничения на кривизну построенного пути и требование непрерывности кривизны вдоль пути, ограничения на кручение кривой, задающей путь, и др.

Для получения траекторий, удовлетворяющих дополнительным ограничениям, применяют два основных подхода:

- получение опорного пути неспециализированным методом и последующее его сглаживание и модификация;
- использование метода, позволяющего учитывать заданные ограничения.

Группа методов, реализующих первый подход, включает различные способы сглаживания, в том числе с использованием клотоид, В-сплайнов и кривых Безье [23].

Группа методов, использующая второй подход, немногочисленна. В основном эти методы решают задачу о посещении всех указанных контрольных точек при наличии ограничений на свойства пути или траектории. В последние годы появляются работы, где решаются обе задачи: учет препятствий и прокладывание траектории с заданными свойствами [27].

Существуют алгоритмы на графах, позволяющие получить менее извилистый (в смысле числа поворотов) путь. Однако эти алгоритмы более трудоемки по сравнению со стандартными, поскольку на каждом шаге просматривают не только вершины, непосредственно

достижимые из текущей вершины (включенные в ее список смежности), но и более удаленные от текущей. Это увеличивает объем используемой памяти и время вычисления. К упомянутым алгоритмам относятся $A^*[PS]$ (A^* Post-Smoothened), Θ^* и $S\text{-}\Theta^*$ [24].

Если алгоритм поиска поддерживает нерегулярные замощения пространства, для получения путей более высокого качества можно выделять несодержащие препятствий выпуклые области, содержащие множество первичных ячеек. Это обеспечит большую свободу при прокладывании пути, хотя такая кластеризация потребует предварительной обработки карты.

Известны также подходы [25] к сглаживанию пути, полученному с помощью методов, использующих для нахождения допустимого пути построение случайных деревьев (RRT — Rapidly Exploring Random Tree). Данный подход основан на диаграммах Вороного и хорошо подходит для решения задач высокой размерности.

Заключение

Отметим одну интересную особенность: многие современные алгоритмы поиска допустимых путей разрабатывались в первую очередь для применения в компьютерных играх. С одной стороны, такие алгоритмы хорошо протестированы и работают достаточно быстро. С другой стороны, для оценки возможностей их использования для реальных объектов требуется дополнительное исследование.

В большинстве упомянутых в настоящем обзоре публикаций рассматриваются двухмерные задачи поиска допустимого пути на плоскости. Практическая применимость таких алгоритмов для проведения расчетов в случае трехмерной среды с препятствиями требует дополнительного исследования.

При переходе к трехмерной задаче прежде всего возникает проблема конечности вычислительных ресурсов. При использовании алгоритмов поиска на графах эта проблема проявляется в невозможности хранения большого объема информации и существенных вычислительных затратах.

Так, если рассматривать покрытие области поиска кубическими ячейками, то каждая такая ячейка имеет в трехмерном случае двадцать шесть соседей, в то время как при рассмотрении плоской задачи и покрытии области квадратами каждая ячейка имеет всего восемь соседей. Соответственно, граф, построенный с использованием отношения непосредственной достижимости, получается существенно большего «размера», и алгоритмы поиска в среднем будут работать существенно медленнее.

Для упрощения задачи можно ограничиться только рассмотрением соседних ячеек, имеющих с текущей ячейкой общую грань, что уменьшит количество вершин в списке смежности до шести в трехмерном случае. Однако при этом возникает проблема сглаживания полученного пути и проблема его оптимальности по различным критериям.

Также не вполне ясно, можно ли эффективно распространить алгоритмы типа НГА* и АНГА*, предлагающие способ огибания препятствий в двухмерном пространстве на основе анализа областей видимости, на случай трехмерного пространства.

Дополнительную информацию по методам поиска можно найти в обзоре [26]).

Авторы выражают благодарность Алдошину Д.В., оказавшему существенную помощь в подборе материалов.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта 12-07-329а.

Список литературы

1. Гребеников А.Г., Мялица А.К., Парфенюк В.В., Парфенюк О.И., Удовиченко С.В. Общие виды и характеристики беспилотных летательных аппаратов. Справочное пособие. Харьков: Нац. аэрокосм. ун-т «Харьк. авиац. ин-т», 2008. 377 с.
2. Weintraub S. Google buys flying drones // CNN Money / Fortune. Электрон. журн. 2010. Режим доступа: <http://tech.fortune.cnn.com/2010/08/09/google-buys-flying-drones> (дата обращения 07.12.2014).
3. Яковлев К.С., Баскин Е.С. Графовые модели в задаче планирования траектории на плоскости // Искусственный интеллект и принятие решений. 2013. № 1. С. 5–12. Режим доступа: http://aidt.ru/index.php?option=com_content&view=article&id=314:314&catid=146:2013-03-13-14-04-00&Itemid=150&lang=en (дата обращения 07.12.2014).
4. Nehme C.E., Cummings M.L., Crandall J.W. A UAV Mission Hierarchy // MIT Department of Aeronautics and Astronautics, for Charles River Analytics, 2006. 16 p. Режим доступа: http://web.mit.edu/aeroastro/labs/halab/papers/HAL2006_09.pdf (дата обращения 07.12.2014).
5. Кирильченко А.А., Колганов М.А., Платонов А.К. Метод потенциалов в задаче выбора пути: история и перспективы // Препринт ИПМ им. Келдыша. М.: 2001. № 40. Режим доступа: http://library.keldysh.ru/prep_vw.asp?pid=1708 (дата обращения 07.12.2014).
6. Vadakkepat P., Chen Tan K., Ming-Liang W. Evolutionary Artificial Potential Fields and Their Application in Real Time Robot Path Planning // Proceedings of the 2000 IEEE Congress on Evolutionary Computation, vol. 1. 2000. P. 256–263. DOI: [10.1109/CEC.2000.87030](https://doi.org/10.1109/CEC.2000.87030)
7. Connolly C.I., Burns J.B., Weiss R. Path planning using Laplace's equation // Proceedings of IEEE Conference on Robotics and Automation. 1990. P. 2102–2106. DOI: [10.1109/ROBOT.1990.126315](https://doi.org/10.1109/ROBOT.1990.126315)
8. Mei H., Tian Y., Zu L. A Hybrid Ant Colony Optimization Algorithm for Path Planning of Robot in Dynamic Environment // International Journal of Information Technology. 2006. Vol. 12, no. 3. P. 78–88. Режим доступа: http://www.intjit.org/journal/volume/12/3/123_8.pdf (дата обращения 07.12.2014).

9. Rimón E., Koditschek D.E. Exact robot navigation using artificial potential functions // IEEE Trans. on Robotics and Automation. 1992. Vol. 8, no. 5. P. 501–518. DOI: [10.1109/70.163777](https://doi.org/10.1109/70.163777)
10. Ruchti J., Senkbeil R., Carroll J., Dickinson J., Holt J., Biaz S. UAV Collision Avoidance Using Artificial Potential Fields // Technical report #CSSE11-03. Auburn University. 2001. 21 p. Режим доступа: http://www.eng.auburn.edu/files/acad_depts/csse/csse_technical_reports/csse11-03.pdf (дата обращения 07.12.2014).
11. Wang Y., Chirikjian G.S. A New Potential Field Method for Robot path Planning // Proceedings of 2000 IEEE International Conference on Robotics and Automation. 2000. P. 977–982. DOI: [10.1109/ROBOT.2000.844727](https://doi.org/10.1109/ROBOT.2000.844727).
12. Mittal S., Deb K. Three-Dimensional Offline Path Planning for UAVs Using Multiobjective Evolutionary Algorithms // Proceedings of the IEEE Congress on Evolutionary Computation. 2007. P. 3195–3202. DOI: [10.1109/CEC.2007.4424880](https://doi.org/10.1109/CEC.2007.4424880)
13. Kostaras A.N., Nikolos I.K., Tsourveloudis N.C., Valavanis K.P. Evolutionary Algorithm Based On-Line Path Planner for UAV Navigation // Proceedings of the 10th Mediterranean Conference on Control and Automation. 2002. P. 143–150. DOI: [10.1109/TSMCB.2002.804370](https://doi.org/10.1109/TSMCB.2002.804370)
14. Dijkstra E.W. A note on two problems in connexion with graphs // Numerische Mathematik. 1959. Vol. 1. P. 269–271. DOI: [10.1007/BF01386390](https://doi.org/10.1007/BF01386390)
15. Fredman M.L., Tarjan R.E. Fibonacci heaps and their uses in improved network optimization algorithms // 25th Annual IEEE Symposium on Foundations of Computer Science. 1984. P. 338–346. DOI: [10.1145/28869.28874](https://doi.org/10.1145/28869.28874)
16. Knuth D.E. The Art Of Computer Programming. Vol. 1. 3-rd ed. Boston: Addison-Wesley, 1997. 634 p. Режим доступа: <http://www-cs-faculty.stanford.edu/uno/taocp.html> (дата обращения 07.12.2014).
17. 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. No. 2. P. 100–107. DOI: [10.1109/TSSC.1968.300136](https://doi.org/10.1109/TSSC.1968.300136)
18. Russell S. Efficient memory-bounded search methods // In: Neumann B. Proceedings of the 10th European Conference on Artificial intelligence. New York: John Wiley and Sons. 1992. P. 1–5. Режим доступа: <http://www.cs.berkeley.edu/russell/papers/ecai92-sma.ps> (дата обращения 07.12.2014).
19. Kaindl H., Khorsand A. Memory-bounded Bidirectional Search // Proceedings of Association for the Advancement of Artificial Intelligence. 1994. P. 1359–1364. Режим доступа: <http://www.aaai.org/Papers/AAAI/1994/AAAI94-209.pdf> (дата обращения 07.12.2014).
20. Botea A., Müller M., Schaeffer J. Near Optimal Hierarchical Path-Finding // Journal of Game Development. 2004. Vol. 1, no. 1. P. 7–28. Режим доступа: <http://abotea.rsise.anu.edu.au/data/hpastar.pdf> (дата обращения 07.12.2014).

21. Kring A., Champandard A.J., Samarin N. DHPA* and SHPA*: Efficient Hierarchical Pathfinding in Dynamic and Static Game worlds // Proceedings of the 6th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment. 2010. P. 39–44. Режим доступа: <http://www.aaai.org/ocs/index.php/AIIDE/AIIDE10/paper/viewFile/2131/2543> (дата обращения 07.12.2014).
22. Guimaraes Macharet D., Alvez Neto A., Montenegro Campos M.F. Feasible UAV Path Planning Using Genetic Algorithms and Bézier Curves // Proceedings of the 20th Brazilian conference on Advances in artificial intelligence. Berlin: Springer-Verlag Berlin, 2010. P. 223–232. DOI: [10.1007/978-3-642-16138-4_23](https://doi.org/10.1007/978-3-642-16138-4_23)
23. Yang K., Sukkarieh S. Real-time Continuous Curvature Path Planning of UAVs in Cluttered Environments // Proceeding of the 5th International Symposium on Mechatronics and its Applications (ISMA08). 2008. P. 1–6. DOI: [10.1109/ISMA.2008.4648836](https://doi.org/10.1109/ISMA.2008.4648836)
24. Nash A., Daniel K., Koenig S., Felner A. Theta*: Any-angle path planning on grids // Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). 2007. P. 1177–1183. Режим доступа: <http://idm-lab.org/bib/abstracts/papers/aaai07a.pdf> (дата обращения 07.12.2014).
25. De Jesus C.A. Human-Automation Collaborative RRT for UAV Mission Path Planning // Master thesis in Electrical Engineering and Computer Science. The Massachusetts Institute of Technology. 2010. 111 p. Режим доступа: <http://web.mit.edu/aeroastro/labs/halab/papers/americoThesis.pdf> (дата обращения 07.12.2014).
26. De Filippis L., Guglieri G., Quagliotti F. Path Planning Strategies for UAVS in 3D Environments // Journal of Intelligent and Robotic Systems. 2012. Vol. 65, no. 1-4. P. 247–264. DOI: [10.1007/s10846-011-9568-2](https://doi.org/10.1007/s10846-011-9568-2)
27. Pan J., Zhang L., Manocha D. Collision-free and smooth trajectory computation in cluttered environments // The International Journal of Robotics Research. 2012. Vol. 31, no. 10. P. 1155–1175. DOI: [10.1177/0278364912453186](https://doi.org/10.1177/0278364912453186)