

## Анализ алгоритмов планирования движений робота-манипулятора

М.О. Клименко<sup>1</sup>, К.С. Яковлев<sup>2</sup>

<sup>1</sup> Московский физико-технический институт (национальный исследовательский университет)

<sup>2</sup> Федеральный исследовательский центр «Информатика и управление» РАН

В работе рассматривается стандартная задача планирования движений для робота-манипулятора. Состоит она в том, что у робота есть некоторое начальное и конечное состояния, и необходимо перейти из стартового состояния в конечное, избегая столкновений с препятствиями. Для манипулятора эта задача осложняется тем, что пространство состояний обычно имеет большую размерность.

Существует много различных подходов к этой задаче, как связанных с обучением с подкреплением и глубоким обучением, так и алгоритмических. В работе рассматриваются именно алгоритмические подходы, а именно sampling-based алгоритмы. Алгоритмы такого типа некоторым образом случайно выбирают точки из пространства свободных состояний робота (состояний, в которых нет коллизий между роботом и окружающей средой), строят в пространстве состояний граф с вершинами в выбранных точках (ребро между вершинами  $a$  и  $b$  в таком графе значит, что все состояния из отрезка  $ab$  свободны, из-за чего робот спокойно может перейти из одного состояния в другое) и ищут путь на нём.

Большое преимущество таких алгоритмов в том, что время их работы не сильно зависит от размерности пространства состояний, из-за чего они хорошо работают в пространствах высоких размерностей.

В процессе исследования написано ПО для тестирования различных sampling-based алгоритмов планирования в окружении MuJoCo [6] (Ссылка на репозиторий).

Для сравнения алгоритмов проведены эксперименты над 3-DoF роботом с пятью алгоритмами - RRT [1], RRT\* [2], Informed RRT\* [3], BIT\* [4] и FMT\* [5]. Состояли они из большого набора различных задач (под задачей подразумевается пара из начального и конечного состояния на карте, между которыми нужно найти путь) на 7 различных окружениях с препятствиями. Запускались они на 10 с., так как алгоритмы зависят от случайного семплирования, на каждой задаче они запускались по 10 раз. При этом для каждой задачи считалась доля успешных запусков и средняя длина найденного пути. Результаты экспериментов приложены в таблицах 1 и 2.

По результатам экспериментов можно сказать, что для нахождения произвольного пути выгоднее всего использовать FMT\* - этот алгоритм находил путь во всех задачах, доля его успешных попыток построения пути самая большая среди испытывавшихся алгоритмов. Если же необходимо минимизировать путь, то лучше всего использовать BIT\*. Конечно, путь, найденный Informed RRT\* зачастую будет короче, чем найденный BIT\*, но разница между их длинами не сильно большая, а при этом вероятность того, что BIT\* найдёт путь на произвольной задаче гораздо выше, чем у Informed RRT\*.

Таблица 1 : Процент удачных попыток нахождения пути для окружений

Номер окружения	1	2	3	4	5	6	7
RRT	100%	100%	98%	100%	94%	76%	48%
RRT*	100%	100%	100%	96%	92%	68%	38%
Informed RRT*	100%	100%	60%	88%	48%	18%	44%
FMT*	100%	100%	72%	98%	100%	98%	92%
BIT*	100%	100%	84%	100%	96%	86%	74%

Таб.2 : Средняя длина найденного пути для окружений

Номер окружения	1	2	3	4	5	6	7
RRT	7.77316	4.768258	3.795838	9.121482	8.874986	8.91894	10.4944
RRT*	7.479312	4.801332	3.415176	9.791804	8.154446	8.7108	10.5328
Informed RRT*	4.569206	3.012748	2.393774	5.206494	4.028702	3.54466	4.681297
FMT*	6.379804	3.09595	3.475242	7.503582	7.260586	7.076946	6.79659
BIT*	4.653224	3.081722	3.891768	4.481956	4.028522	5.817524	5.37925

## СПИСОК ЛИТЕРАТУРЫ

- [1] LaValle, Steven M.. “Rapidly-exploring random trees : a new tool for path planning.” The annual research report (1998).
- [2] Karaman, Sertac and Emilio Frazzoli. “Incremental Sampling-based Algorithms for Optimal Motion Planning.” arXiv:1005.0416 (2010).
- [3] Gammell, Jonathan D. et al. “Informed RRT\*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic.” 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (2014): 2997-3004.
- [4] Janson, Lucas and Marco Pavone. “Fast marching tree: A fast marching sampling-based method for optimal motion planning in many dimensions.” The International Journal of Robotics Research 34 (2013): 883 - 921. arXiv:1306.3532v4
- [5] Gammell, Jonathan D. et al. “Batch Informed Trees (BIT\*): Sampling-based optimal planning via the heuristically guided search of implicit random geometric graphs.” 2015 IEEE International Conference on Robotics and Automation (ICRA) (2014): 3067-3074. arXiv:1405.5848v7
- [6] Todorov, Emanuel et al. “MuJoCo: A physics engine for model-based control.” 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (2012): 5026-5033.