

# АЛГОРИТМ ПЛАНИРОВАНИЯ ДВИЖЕНИЯ РОБОТА В НЕОПРЕДЕЛЕННОЙ ДИНАМИЧНОЙ СРЕДЕ

А. В. Господ

Кафедра автоматизации технологических процессов и производств,  
Могилевский государственный университет продовольствия  
Могилев, Республика Беларусь  
E-mail: gospod\_andrei@mail.ru

*Планирование движения робота в условиях недетерминированной информации о внешней среде часто принимает допущения о известности геометрии препятствия и возможности предсказать траекторию движения экстраполяцией, что в реальности невозможно. В [1] был введен алгоритм collision-free receiver (CFR – бесконтактная воспринимающая), который может обнаружить траектории свободные от столкновений при известных геометрии или движении. В этой работе рассматривается как использовать CFR в режиме реального времени, на работе с  $n$  степенями свободы, при неизвестных траектории и размерах препятствий, минимизируется количество опасных остановок, когда робот мог бы столкнуться с объектом.*

Планирование движения роботов в неопределенной динамичной среде привлекает все больше внимания в исследовательской робототехнике. Одно из известных предположений, ничего не имеющее общего с реальностью, о том, что движение препятствия известно [2,3]. Второе предположение – определена геометрия препятствия. Если движение неизвестно, то используется общий подход для прогнозирования будущего движения, отслеживая прошлое [4,5]. Таким образом, на столкновение робота с конфигурацией  $q$  в будущем времени  $t$ , указывает проверка пересечения конфигурации робота  $(q, t)$  с каждой частью препятствия в момент времени  $t$ .

В данной работе предложен новый алгоритм планирования движения робота T-RAMP, включающий в себя улучшение и объединение алгоритмов: RAMP [6] и CFR [1]. Из RAMP взяты адаптивность, а из CFR понятия точечного препятствия (ТП) и динамического пакета (ДП). Добавлено планирование движения робота в реальном времени, при неизвестных геометрии и траектории движения препятствий, и минимизация небезопасных остановок, когда робот может быть сбит другим объектом.

Алгоритм CFR находит точку  $\chi = (q, t)$ , свободную от столкновений, проверяя пересечения между ДП  $E(\chi, t)$  и ТП в каждый момент времени, начиная с  $\tau_0$  и до любого столкновения, или за максимальное время наблюдений  $t - \tau_0$ , который контролируется системной переменной тактового сигнала  $tclock$  (эта переменная изменяется независимо от алгоритма CFR), как показано в алгоритме 1. Интервал между двумя соседними моментами зондирования  $\delta t$ , т.е. частота зондирования  $1/\delta t$ . Каждая итерация цикла, как правило, занимает больше времени, чем  $\delta t$ , следовательно, после каждой итерации есть обновленные данные от датчика для следующей итерации.

Как правило, для робота, состоящего из нескольких звеньев, каждое звено можно аппроксимировать набором геометрических фигур, таких как прямоугольник, квадрат, сфера, параллелограмм. Теперь ДП может быть создан для каждого звена. Так как ДП для всего робота – сумма ДП звеньев, необходимо сосредоточиться на том, как проверить пересечение между ДП звеньев и ТП. Используем экстракцию, чтобы определить ТП. В [7] описана проверка пересечения ДП и ТП (IDEAOS), наиболее эффективный алгоритм заявленный в CFR.

RAMP основан на дискретизации и это особенно эффективно при планировании движения робота с большим количеством степеней свободы в условиях недетерминированной информации о внешней среде из-за следующих особенностей:

- в реальном времени одновременное движение робота и планирование его траектории;
- параллельное планирование с оптимизацией, как в эволюционном алгоритме [8], и постоянное обновление возможных траекторий от его текущей конфигурации к целевой;
- структурная гибкость, что позволяет онлайн адаптацию к различным внешним условиям и офлайн дополнения для различных конфигураций роботов.

Все основные компоненты RAMP настраиваемы. Сила RAMP в его гибкости для адапций и расширений. RAMP всегда имеет набор различных траекторий в пространстве – пакет траекторий. Начальный пакет траекторий может быть сформирован случайно. Каждая траектория начинается с текущей конфигурации робота и заканчивается в целевой, и может быть лишь частично выполнимой – имеет неосуществимый сегмент. Качество траектории, с точки зрения выполнимости и оптимальности, оценено через функцию оценки, объединяющую критерии оптимизации, такие как: кратчайшее время движения, максимальное время выполнимого сегмен-

та и т.д. Как только имеется выполнимый сегмент траектории, RAMP позволяет роботу движение по нему, при этом вычисляя последующие возможные сегменты, таким образом робот может переключиться на более лучший вариант. Три цикла запускаются одновременно:

- сенсорные данные обновляются при каждом зондировании;
- изменение траектории и оценка (переоценка) на основе данных датчика в каждом цикле расчета траектории;
- робот переключается с текущей на лучшую траекторию в цикле контроля или адаптации.

RAMP определяется эффективным онлайн обнаружением выполнимых траекторий. Первоначальный RAMP предполагает, что геометрия препятствий известна, и проводит проверку на основе предсказания движения. Это реально только при моделировании.

Используя CFP в RAMP для проверки сегмента траектории (посредством обнаружения свободных от столкновений), можно устранить нереалистичное предположение об известности геометрии препятствия и недостаток в прогнозировании траектории движения препятствия. Сегмент траектории свободный от столкновений, найденный CFP, по которому движется робот обозначается  $\Gamma_1$ , а в это время RAMP ищет следующую возможную траекторию. После того как робот прошел сегмент  $\Gamma_1$ , если найден следующий безопасный сегмент траектории  $\Gamma_2$ , он продолжает плавное движение по нему.

И все же CFP необходимо конечное время, чтобы обнаружить свободные от столкновений точки, по которым пройдет траектория движения робота. Время обнаружения свободных точек  $\chi = (q, t)$  зависит от двух факторов:

- 1) размер динамического пакета  $E(\chi, t)$ , который зависит от  $v_{max}(t - \tau)$ , который уменьшается со временем  $\tau$ ;
- 2) вычислительные мощности компьютера и датчиков.

Первый фактор, как правило, доминирует. Пусть  $\tau_0$  время начала наблюдений и проверки, свободна ли точка  $(q, t)$  от столкновений. Если  $E(\chi, t)$  свободен от точечных препятствий при  $\tau_1 < t$ , то CFP необходимо  $(\tau_1 - \tau_0)$  времени, чтобы найти  $\chi$  свободную от столкновений. Поэтому следующий выполнимый сегмент траектории не может быть найден при завершении роботом текущего сегмента, приводящего к остановке. Во время такой принудительной остановки робот может быть сбит препятствием. Поэтому важное расширение RAMP состоит в минимизации остановок. Параллелизм и гибкость RAMP добивается следующим:

- Добавление к функции оценки, в качестве дополнительного критерия оптимизации, увеличение безопасного времени  $\delta t_{safe}$ , которое он проводит в сегменте траектории без столкнове-

ний. RAMP выберет допустимый сегмент траектории (среди всех найденных возможных), который максимизирует общее время  $\Delta t_{safe} = \Delta t_{move} + \delta t_{safe}$ , где  $\Delta t_{move}$  время движения в сегменте. Примечательно, что  $\delta t_{safe}$  в точке  $\chi_e = (q_e, t_e)$  не может быть известно точно, прежде чем робот достигнет  $\chi_e$ , но его можно оценить с помощью  $d_{min}(q_e, \tau)/v_{max} - (t_e - \tau)$  для  $\tau < t_e$ , где  $d_{min}(q_e, \tau)$  – минимальное расстояние между роботом и ТП. Поскольку  $\delta t_{safe}$  рассчитывается для точки  $\chi_e$  без столкновений, при  $\tau > t_e$ . Разработан метод расчета  $d_{min}(q_e, \tau)$  между точечными препятствиями вблизи робота в точке  $\chi_e$ .

- Отделением проверки столкновений (CFP) от оценки пригодности траектории. Проверка столкновений постоянно происходит в фоновом режиме и дает информацию о свободной траектории, в то время как блок оценки пригодности просто использует эту информацию для вычисления значения функции пригодности, что гораздо быстрее, чем ранее. Назовем расширенный алгоритм T-RAMP, алгоритм 2.

В алгоритме 2 есть четыре одновременных шага, проверка на столкновение, планирование, адаптация и движение. Каждый цикл адаптации состоит из многократных циклов планирования. Каждая итерация цикла зондирования длится  $\delta \tau$ , которая определяется проектировщиком, таким образом новая информация полученная при каждом зондировании используется в CFP (алгоритм 1), который в свою очередь вызывается в алгоритме 3, для проверки столкновения.

1. Vatcha, R., Xiao, J.: Perceived CT-space for motion planning in unknown and unpredictable environments. In: Intl. Workshop on the Algorithmic Foundations of Robotics, WAFR (December 2008)
2. Fiorini, P., Shiller, Z.: Motion planning in dynamic environments using velocity obstacles. Intl. J. of Robotics Research, 760–772 (1998)
3. Large, F., Sckhavat, S., Shiller, Z., Laugier, C.: Using non-linear velocity obstacles to plan motions in a dynamic environment. In: IEEE Intl. Conf. on Control, Automation, Robotics and Vision (ICARCV), pp. 734–739 (2002)
4. Gallagher, G., Srinivasa, S.S., Bagnell, J.A., Ferguson, D.: Gatmo: a generalized approach to tracking movable objects. In: IEEE Intl. Conf. on Robotics and Automation, pp. 2043–2048 (May 2009)
5. Ess, A., Leibe, B., Schindler, K., Gool, L.V.: Moving obstacle detection in highly dynamic scenes. In: IEEE Intl. Conf. on Robotics and Automation, pp. 56–63 (May 2009)
6. Vannoy, J., Xiao, J.: Real-time Adaptive Motion Planning (RAMP) of mobile manipulators in dynamic environments with unforeseen changes. IEEE Trans. on Robotics 24(5), 1199–1212 (2008)
7. Vatcha, R., Xiao, J.: An efficient algorithm for online determination of collision-free configuration-time points directly from sensor data. In: IEEE Intl. Conf. on Robotics and Automation (May 2010)
8. Bonissone, P.P., Subbu, R., Eklund, N., Kiehl, T.R.: Evolutionary algorithms + domain knowledge = real-world evolutionary computation. IEEE Trans. Evolutionary Computation 10(3), 256–280 (2006)