УДК 004.896

DOI 10.35211/1990-5297-2020-9-244-39-43

*В. В. Изжболдина, И. В. Ватаманюк, Е. Е. Усина***АЛГОРИТМ НАЗНАЧЕНИЯ ЦЕЛЕВЫХ ТОЧЕК
ДЛЯ БЕСКОЛЛИЗИОННОЙ РЕКОНФИГУРАЦИИ РОЯ РОБОТОВ****Санкт-Петербургский институт информатики и автоматизации
Российской академии наук
vatamaniuk.i.v@gmail.com**

Представлен алгоритм реконфигурации роя роботов без коллизий, основанный на методе поиска коротких соответствий между стартовыми и целевыми позициями, который минимизирует суммарную длину прямолинейных траекторий роботов. Представлены результаты моделирования для роя размером до 10 000 агентов.

Ключевые слова: алгоритм реконфигурации, роевая робототехника, мобильные роботы, групповое управление, реконфигурация.

*V. V. Izboldina, I. V. Vatamaniuk, E. E. Usina***ALGORITHM OF TARGET POINT SETTING
FOR COLLISION-FREE RECONFIGURATION OF ROBOT SWARM****St. Petersburg Institute for Informatics and Automation
of the Russian Academy of Sciences**

An algorithm for collision-free robot reconfiguration is presented, based on search for short mappings between start and target positions, which minimizes the total length of straight robotic paths. Modeling results for a swarm up to 10000 agents are presented.

Keywords: reconfiguration algorithm, swarm robotics, mobile robots, group control, reconfiguration.

Введение

Решение задачи реконфигурации автономных мобильных роботов в пространстве является основополагающим этапом в задаче управления роем роботов [1]. Под термином «конфигурация» понимается взаимное расположение роботов, к примеру, образующих некоторую фигуру в пространстве. Под реконфигурацией здесь понимается перестроение роя из одной конфигурации в другую. Целью данной статьи является разработка алгоритмов реконфигурации роя однотипных роботов при выполнении совместных задач.

В литературе описано множество версий задачи о реконфигурации роя роботов с различными сценариями, правилами и подходами [2–4]. Большинство алгоритмов реконфигурации роя роботов разработаны для случая, когда пространство ограничено [5]. В серии статей [6–8], рассматривается задача реконфигурации взаимно непересекающихся непронумерованных дисков. Реконфигурация состоит из серии последовательных прямолинейных передвижений (шагов) дисков по отдельности. Задача заключается в минимизации числа шагов. Полученные результаты в этом направлении таковы: 1) для n_d дисков всегда достаточно $2n_d - 1$ шага;

2) задача о реконфигурации в модели переносов имеет nP -сложность; 3) для некоторых конфигураций $(5n_d / 3) - 1$ является необходимым числом шагов, но при этом невозможно получить реконфигурацию без коллизий (столкновений объектов при передвижении) с исключительно прямолинейными траекториями для всех возможных конфигураций.

Предлагаемые в данном исследовании алгоритмы реконфигурации предполагают одновременное движение роботов к своим целевым точкам без коллизий. При этом разработанные алгоритмы позволяют сократить число перемещений (шагов) и снизить энергопотребление робота от аккумулятора [9]. Назначаемые роботам траектории при этом прямолинейны, а предлагаемое решение обладает вычислительной сложностью, приемлемой для применения в рое с большим количеством роботов.

**Разработка алгоритма
реконфигурации роя роботов**

Задача реконфигурации роя была представлена в следующем виде: необходимо равномерно расположить множество роботов $A = \{a_1, a_2, \dots, a_i, a_j, \dots, a_N\}$ в геометрической фигуре T на плоскости. Стартовая конфигурация

роя произвольна и представлена множеством точек $S = \{s_1, \dots, s_N\}$. Необходимо разбить заданную фигуру на целевые точки T в соответствии с заданной плотностью размещения роботов D . Конечная конфигурация роя представлена набором точек $T = \{t_1, \dots, t_N\}$. Соответствие между элементами множеств S и T является произвольной биекцией между конфигурациями $(s_i \rightarrow t_i)$. Суммарной длиной соответствия называется сумма длин всех отрезков, соединяющих стартовые и целевые точки:

$$TM = \sum_{i=1}^N |s_i t_i|.$$

Соответствие между множествами S и T считается коротким, если его суммарная длина TM минимальна. Алгоритм реконфигурации для конфигураций S и T есть кусочно-линейная гомотопия в классе допустимых (без коллизий) конфигураций, переводящая S в T . В каждый момент времени конфигурация должна быть допустимой. Соответствие между множествами S и T является свободным, если расстояние между любыми двумя отрезками $[s_i t_i]$ и $[s_j t_j]$ не

меньше $d_{\min}$ при $i \neq j$. Свободное соответствие обеспечивает одновременное движение роботов к целевым точкам по прямолинейным траекториям без коллизий. Пример несвободного соответствия показан на рис. 1, где черные точки обозначают стартовую конфигурацию роботов, а белые – целевую конфигурацию.

Рис. 1. Случай несвободного соответствия

Рис. 1 показывает, что в случае, если роботов всего два, свободное соответствие существует не всегда. Проблема несвободного соответствия может быть решена при помощи назначения агентам задержек.

Если существует коллизия роботов a_i и a_j , данная пара роботов называется зависимой (рис. 2).

Рис. 2. Зависимые пары для разных случаев коллизии:
а – коллизия в целевой точке; б – коллизия в стартовой точке;
в – косая коллизия; г – коллизия при движении от стартовой к целевой точке

Так как движение агентов роя из стартовых в целевые точки должно осуществляться без коллизий, при разработке алгоритма реконфигурации были использованы следующие методы и приемы: параллельный перенос, гомотетия, обмен целевыми точками, построение графа коллизий, «разделяй и властвуй». В качестве входных параметров алгоритм реконфигурации использует: количество роботов в рое N ; минимальное допустимое расстояние между центрами роботов $d_{\min}$; скорость движения роботов v ; плотность размещения роботов на площади заданной фигуры D .

Минимальное количество агентов в рое равняется двум. Каждый робот имеет форму круга с заданным радиусом R . При этом считается, что N роботов достаточно для покрытия поверхности с заданной плотностью D [9]. Под

плотностью D понимается количество роботов, расположенных на поверхности площадью 10×10 относительных единиц на расстоянии друг от друга, не превышающем минимальное допустимое: $d > d_{\min}$. Предполагается, что при сближении роботов на расстояние меньше $d_{\min}$ происходит коллизия роботов. Плотность и минимальное допустимое расстояние связаны соотношением:

$$\frac{\sqrt{D}}{d_{\min}} < \frac{1}{2}.$$

Также было сделано допущение, что роботы движутся прямолинейно и равномерно со скоростью v . Выходными параметрами алгоритма являются: время реконфигурации роя; траектории движения роботов к целевым точкам. Алгоритм реконфигурации должны обес-

печивать построение траектории движения роботов к целевым точкам без коллизий, при этом суммарная длина траектории должна быть минимальной.

Как можно заметить, при увеличении расстояния между центральными точками S^* и T^* потенциальное число коллизий также увели-

вается. Поэтому для реконфигурации целесообразно сначала переместить конфигурацию S в положение S' , а затем конфигурацию S' – в T . Для этого может быть использован метод параллельного переноса, который предполагает перенос множества S таким образом, чтобы точки S^* и T^* совпали (рис. 3).

Рис. 3. Конфигурация роя до (а), во время (б) и после (в) использования параллельного переноса

Параллельный перенос позволяет достичь свободного короткого соответствия. Метод обмена целевыми координатами точек конечной конфигурации дает пару роботов, которая больше не имеет коллизий. Примеры устранения коллизий при движении вдоль отрезка с применением метода обмена целевыми точками показаны на рис. 4.

Рис. 4. Примеры устранения коллизии вдоль отрезка методом обмена целевых точек

При большом количестве агентов в рое дополнительно могут быть использованы такие методы и приемы, как построение графа коллизий, назначение задержек и «разделяй и властвуй».

Граф коллизий состоит из вершин (стартовые позиции роботов представлены в виде точек – вершин графа) и ребер, соединяющих пары роботов с коллизией. Пример построения графа коллизий представлен на рис. 5. Сначала находим короткое соответствие между стартовыми и конечными положениями роботов (рис. 5, а). Стрелками обозначено направление движения роботов к целевым точкам. Далее устраняем сильно зависимую пару методом обмена целевыми точками (рис. 5, б). На основе полученной информации построим граф коллизий (рис. 5, в).

Рис. 5. Пример построения графа коллизий

Назначение задержек роботам в стартовых позициях дает свободную от коллизий реконфигурацию, если прямолинейное движение

к целевым точкам с предписанными задержками не создает коллизий. Задержка представляет из себя рассчитанное особым образом ин-

тервал времени для каждого робота с коллизией, которое откладывает начало движения робота на этот интервал. То есть движение робота начнется позже или, говоря другими словами, с задержкой по времени. В итоге роботы двигаются прямолинейно непосредственно к целевой точке с одной и той же скоростью, начав движение с предписанными задержками.

Идея приема «разделяй и властвуй» такова: происходит деление конфигурации на независимые области, и реконфигурация происходит в каждой из областей независимо от других и, в то же время, синхронно с другими. Подробнее с этими методами можно ознакомиться в [10]. Разработанный алгоритм представлен на рис. 6.

Рис. 6. Разработанный алгоритм реконфигурации роя

Разработанный алгоритм (рис. 6) включает в себя метод параллельного переноса, метод назначения задержек, построение графа коллизий и алгоритм «разделяй и властвуй». В начале работы алгоритма определяются центры стартовой S и целевой T конфигураций. При помощи метода параллельного переноса совмещаются центральные точки S^* и T^* . Затем вычисляются конфигурации S' и T' с общей центральной точкой, а также короткое соответствие между конфигурациями. После этого происходит построение графа коллизий и назначение агентам задержек. Далее проводится проверка на наличие коллизий между агентами роя. В случае, если коллизии сохраняются, применяется алгоритм «разделяй и властвуй» до тех пор, пока все коллизии не будут устранены.

Результаты экспериментов и их обсуждение

Для оценки работы предложенного алгоритма была разработана компьютерная модель, где конечной конфигурацией является квадрат с центром в точке $(0, 0)$. Скорость перемещения движения роботов была установлена $v = 1$ м/с. Радиус робота был задан $R = 1$ (в относительных единицах). Была рассмотрена такая конфигурация, при которой расстояние между стартовыми и целевыми координатами роботов составляло не менее двух длин диагонали квадрата T (60 единиц в данном эксперименте). Для оценки среднего времени вычисления траекторий движения была проведена серия экспериментов с роем размером 10, 100, 1000, 10000 роботов. Длина стороны квадрата T была рассчи-

тана исходя из плотности размещения роботов в каждом случае отдельно. Результаты тестирования показаны на рис. 7.

Рис. 7. Среднее время вычислений траекторий роботов

Разработанный алгоритм не может быть использован для больших роев ($N > 100$) при решении задач реконфигурации в режиме реального времени, однако может с успехом применяться для предварительного расчета траекторий роботов, а также в реальном времени для роя с относительно небольшим количеством агентов (до 100). Разработанный алгоритм имеет высокую сложность – $O(n^3 \cdot \log n)$ по причине использования приема «разделяй и властвуй», который имеет временную сложность $O(n \cdot \log n)$. Также следует отметить, что назначаемые роботам прямолинейные траектории минимизирует суммарную длину перемещений, что способствует сокращению энергопотребления робота от аккумулятора.

Заключение

Разработанный алгоритм реконфигурации роя может быть использован совместно с классическими подходами, а также интегрирован в другие алгоритмы реконфигурации. Основным достоинством алгоритма является возможность одновременного движения роботов к целевым точкам по прямолинейным траекториям, что позволяет уменьшить суммарную длину

траекторий. Дальнейшие исследования будут направлены на усовершенствования разработанных алгоритмов для возможности их использования в реальных условиях. Авторы ожидают, что наилучшие практические результаты будут получены при сочетании предложенных алгоритмов с методом виртуальных взаимно отталкивающих сил между роботами, расположенных друг от друга на малых расстояниях.

Благодарим д-ра физ.-мат. наук Гаянэ Юрьевну Панину за сотрудничество, ценные замечания и экспертное содействие в данном исследовании.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Shlyakhov, N. E. Survey of methods and algorithms of robot swarm aggregation / N. E. Shlyakhov, I. V. Vatamaniuk, A. L. Ronzhin // Journal of Physics: Conference Series. – 2017. – Т. 803.
2. Chen, Y. Q. Formation control: a review and a new consideration / Y. Q. Chen, Z. Wang // 2005 IEEE/RSJ International conference on intelligent robots and systems. – IEEE, 2005. – С. 3181–3186.
3. Hsieh M. A. Decentralized controllers for shape generation with robotic swarms / M. A. Hsieh, V. Kumar, L. Chaimowicz // Robotica. – 2008. – Т. 26, № 5. – С. 691–701.
4. Yan, X. Multilevel-based topology design and shape control of robot swarms / X. Yan, J. Chen, D. Sun // Automatica. – 2012. – Т. 48, № 12. – С. 3122–3127.
5. Adler, A. Efficient Multi-Robot Motion Planning for Unlabeled Discs in Simple Polygons / A. Adler, M. de Berg, D. Halperin, K. Solovey // IEEE T. Automation Science and Engineering. – 2015. – Т. 12, № 4. – С. 1309–1317.
6. Abellanas, M. Moving coins / M. Abellanas, S. Bereg, F. Hurtado, A. G. Olaverri, D. Rappaport, J. Tejel // Computational Geometry. – 2006. – Т. 34, № 1. – С. 35–48.
7. Bereg, S. Sliding disks in the plane / S. Bereg, A. Dumitrescu, J. Pach // International Journal of Computational Geometry & Applications. – 2008. – Т. 18, № 5. – С. 373–387.
8. Dumitrescu, A. On reconfiguration of disks in the plane and related problems / A. Dumitrescu, M. Jiang // Computational Geometry. – 2013. – Т. 46, № 3. – С. 191–202.
9. Vatamaniuk, I. V. Reconfiguration of robot swarm formation / I. V. Vatamaniuk, G. Y. Panina, A. L. Ronzhin // Upravlenie Bol'shimi Sistemami. – 2015. – Т. 58. – С. 285–305.
10. Izhboldina, V. Approach to UAV Swarm Control and Collision-free Reconfiguration / V. Izhboldina, I. Lebedev, A. Shabanova // Proceedings of 15th International Conference on Electromechanics and Robotics "Zavalishin's Readings" DOI: 10.1007/978-981-15-5580-0 (In press).