

Д. Е. Кузаков¹, М. С. Дьяков², М. М. Лаврентьев³

¹ Новосибирский государственный университет
ул. Пирогова, 2, Новосибирск, 630090, Россия

² ООО «СофтЛаб-НСК»
пр. Акад. Коптюга, 1, Новосибирск, 630090, Россия

³ Институт автоматизации и электрометрии СО РАН
пр. Акад. Коптюга, 1, Новосибирск, 630090, Россия

kuzd@sl.iae.nsk.su, diam@sl.iae.nsk.su, mmlavrentiev@gmail.com

ПОИСК ПУТЕЙ ДЛЯ ГРУППЫ АВТОНОМНЫХ ТРАНСПОРТНЫХ СРЕДСТВ ПРИ ИССЛЕДОВАНИИ НЕИЗВЕСТНОЙ ТЕРРИТОРИИ

Представлен алгоритм исследования заранее неизвестной территории с помощью группы автономных транспортных средств (АТС). В нем используется новый метод выбора точек назначения для каждого АТС из группы. Новизна данного метода заключается в использовании кластеризации граничной области – области карты препятствий, находящейся на границе ее исследованной части. Каждому АТС сопоставляется некоторый кластер с помощью поиска паросочетания минимального веса. Точка назначения выбирается из кластера с помощью функции приоритета – функции, определяющей выгодность выбора клетки в зависимости от затрат на ее достижение, количества полученной информации и расстояния до целей других АТС.

Ключевые слова: исследование территории, кластеризация граничной области, гранично-ориентированный алгоритм, группа автономных транспортных средств.

Введение

В настоящее время применение автономных транспортных средств (АТС) для выполнения различных задач – популярное направление робототехники. Одна из таких задач – исследование местности. Группа АТС может быть использована для исследования опасной для людей территории (поиск радиоактивных веществ, разминирование местности, поиск пострадавших и др.), а также для автоматизации монотонных задач, связанных с исследованием местности (например, для составления карт).

Целью работы является решение задачи исследования территории с помощью группы АТС. Данная задача заключается в следующем. На изначально неизвестной территории расположено несколько АТС. Каждое из АТС оснащено приборами, позволяющими исследовать некоторый участок местности вокруг его текущего положения. На основе данных, получаемых от приборов, строится карта препятствий. Необходимо для каждого АТС построить путь, дви-

Кузаков Д. Е., Дьяков М. С., Лаврентьев М. М. Поиск путей для группы автономных транспортных средств при исследовании неизвестной территории // Вестн. Новосиб. гос. ун-та. Серия: Информационные технологии. 2016. Т. 14, № 2. С. 59–71.

жение по которому позволит разведать участок еще не исследованной территории. Пути должны перестраиваться по мере изменения информации о карте препятствий.

В зависимости от способа обмена информацией между транспортными средствами (ТС) существует два типа задач исследования. При централизованном исследовании [1-4] строится общая карта препятствий на основе данных от всех ТС. При децентрализованном исследовании [5; 6] каждому ТС доступна лишь часть информации (как правило, данные с приборов этого ТС и ближайших к нему других ТС). В работе решается задача централизованного исследования.

Представленный в работе алгоритм рассматривает карту препятствий в виде сетки. Каждая клетка карты является либо исследованной, либо неисследованной. Исследованная клетка может являться препятствием. Не являющаяся препятствием клетка называется свободной. Множество исследованных клеток карты далее называется исследованной областью.

На данный момент существует большое количество алгоритмов решения задачи исследования территории. Существуют различные типы таких алгоритмов.

- Муравьиные алгоритмы [5]. Основаны на максимизации локальной эффективности исследования местности каждым из АТС, что делает их эффективными для использования в децентрализованных системах.

- Алгоритмы покрытия деревьями [1; 2; 7]. Данный тип алгоритмов представляет свободное пространство карты в виде набора деревьев. Вершинам соответствуют клетки карты. Ребра соответствуют траектории движения ТС при исследовании. Каждое из деревьев соответствует одному ТС из группы.

- Алгоритмы потенциального поля [6; 8]. Такие алгоритмы сопоставляют каждой клетке карты препятствий направление, в котором при исследовании должно двигаться ТС, находящееся в данной клетке. Основным недостатком данного подхода является невозможность глобального распределения ТС по неисследованной области.

- Гранично-ориентированные алгоритмы [3; 4; 9–14]. Данный тип алгоритмов описан ниже.

Представленный в работе алгоритм относится к типу гранично-ориентированных алгоритмов. Алгоритмы этого типа опираются на понятие граничной клетки (ГК). Граничной клеткой называется исследованная клетка, при нахождении в которой будет исследован некоторый участок неисследованной части территории. Множество всех граничных клеток составляет граничную область (ГО) карты.

Гранично-ориентированные алгоритмы итеративно выполняют определенную последовательность действий. Каждому свободному ТС (т. е. которому в данный момент не назначен путь) сопоставляется некоторая целевая ГК в качестве точки назначения. После этого ТС начинает движение по пути из текущего положения в заданную ГК. По мере движения ТС исследованная часть карты расширяется, а также меняется и сама граничная область. После завершения движения по заданному пути эта последовательность действий повторяется.

Таким образом, каждое ТС из группы постоянно двигается в некоторую ГК. При этом разные ГК в различной степени полезны для исследования и требуют различные затраты для проезда в них каждым ТС из группы. Выбор точки назначения для ТС выполняется с помощью функции приоритета – функции, определяющей выгодность выбора некоторой ГК для данного ТС. От выбора данной функции зависит эффективность исследования (т. е. затраченное время и суммарная длина пути, пройденного группой ТС при исследовании). В литературе используются функции, обеспечивающие следующие стратегии выбора точки назначения:

- минимизация C (Cost подход) [3; 4; 9];
- максимизация U и минимизация C (Cost-Utility подход) [10; 11];
- максимизация T или T' и минимизация C (Coordinated подход) [12];
- максимизация T , максимизация U и минимизация C (Гибридный подход) [13].

Здесь C – оценка затрат АТС на проезд в ГК (например, длина кратчайшего пути), U – оценка полезности выбора ГК в качестве точки назначения (в [10] используется площадь участка неисследованной области, которая будет зафиксирована приборами; в [11] – концентрация газа

в ГК, в [13] – количество неисследованных клеток вблизи рассматриваемой ГК), T – расстояние от ГК до ближайшей из целей других АТС, T' – расстояние от ГК до текущего положения ближайшего из других АТС.

Некоторые алгоритмы используют разбиение неисследованного пространства на кластеры, которые затем распределяются между ТС [14]. При выборе точки назначения для некоторого ТС в первую очередь рассматриваются те ГК, которые позволяют исследовать часть соответствующего ему кластера.

Формулировка требований к алгоритму

Сформулируем требования к разрабатываемому алгоритму исследования местности.

1. Должна решаться задача централизованного исследования.
2. Работа в условиях постоянного изменения карты препятствий (за счет открытия участков неисследованной области карты).
3. Оптимизация суммарного расстояния, пройденного группой ТС.
4. Распределение ТС по неисследованной области.
5. Возможность использования алгоритма с различными моделями ТС: кинематическими и геометрическими (форма и размер ТС, учитываемые при объезде препятствий).
6. Возможность использования алгоритма при различном поле зрения вокруг ТС (различной форме, размерах исследуемого участка и в зависимости от окружающих препятствий).

В связи с данными требованиями, для разработки выбран класс гранично-ориентированных алгоритмов. Выбор обусловлен следующими обстоятельствами. Муравьиные алгоритмы, как правило, основаны на максимизации локальной эффективности исследования местности каждым из АТС. Это делает их неэффективными для применения к задаче централизованного исследования. Алгоритмы покрытия деревьями выполняют построение полного графа возможных перемещений ТС для каждого набора входных данных. Это требует больших вычислительных затрат при сложной кинематической модели ТС. Алгоритмы потенциального поля при построении поля направлений опираются на локальную информацию о местности. Поэтому в таких алгоритмах сложно (а иногда и невозможно) глобально распределить ТС по неисследованной области карты.

Общая схема алгоритма

Будем называть ТС активным, если для него существует хотя бы одна достижимая ГК. Изначально все ТС считаются активными, но во время работы алгоритма некоторые из них могут стать неактивными. Неактивные ТС являются бесполезными для исследования, поэтому для них не нужно вычислять путь. Кроме того, они не учитываются при вычислении путей для активных ТС.

Будем называть путь для ТС актуальным, если одновременно выполняются следующие условия: 1) ТС не завершило движение по данному пути; 2) движение по назначенному пути даст информацию о неисследованной области.

Разработанный алгоритм является итеративным. Каждая итерация алгоритма принимает в качестве входных данных информацию о карте препятствий и положения ТС на карте. Также используется путь, сопоставленный для каждого ТС на предыдущей итерации (если данная итерация не является первой).

Результатом работы итерации является путь для каждого активного ТС из группы. При этом вычисление нового пути происходит только в случае, если текущий путь перестал быть актуальным. Иначе говоря, если ТС завершило движение по пути, или если движение по нему не даст новой информации о карте препятствий (например, если область вокруг точки назначения была исследована другим ТС).

Итерация алгоритма состоит из следующих частей.

1. Получение данных о текущих положениях ТС и карте препятствий.
2. Проверка актуальности путей, полученных на предыдущей итерации.
3. Расчет путей для каждого активного ТС во все достижимые для него клетки карты.
4. Выявление неактивных ТС.
5. Выбор точек назначения из ГО для всех активных ТС.
6. Построение пути из текущего положения каждого ТС в сопоставленную ему точку назначения.

Алгоритм выбора точек назначения для ТС

Наиболее важной частью алгоритма исследования является алгоритм выбора точек назначения. Данный алгоритм выглядит следующим образом.

1. Выделение и кластеризация ГО.
2. Сопоставление каждому ТС одного из полученных кластеров.
3. Выбор точки назначения для каждого ТС из соответствующего ему кластера.

Выделение и кластеризация ГО. Для выделения ГО рассматриваются все свободные клетки. На каждую из них накладывается шаблон поля зрения ТС. Если в него попала неисследованная клетка, точка сохраняется в качестве граничной.

Для того чтобы распределить ТС по неисследованной территории, используется разбиение ГО на кластеры по количеству активных ТС. Разбиение ГО было выбрано вместо разбиения неисследованной области карты по определенным причинам. Во-первых, при кластеризации неисследованной области часто выделяются кластеры, не граничащие с исследованной областью. В этом случае невозможно построить гарантированно проезжаемый путь для ТС для исследования кластера. При этом граничная область всегда может быть разбита на кластеры, каждый из которых достижим хотя бы одним ТС из группы. Во-вторых, граничная область часто содержит меньше клеток, чем неисследованная область, поэтому кластеризация ГО выполняется быстрее.

Разработанный алгоритм кластеризации основан на алгоритме K-means [15]. В качестве координат клеток, необходимых алгоритму K-means, используются их целочисленные координаты на карте препятствий. В качестве метрики берется евклидово расстояние. Пример кластеризации граничной области при исследовании территории приведен на рис. 1.

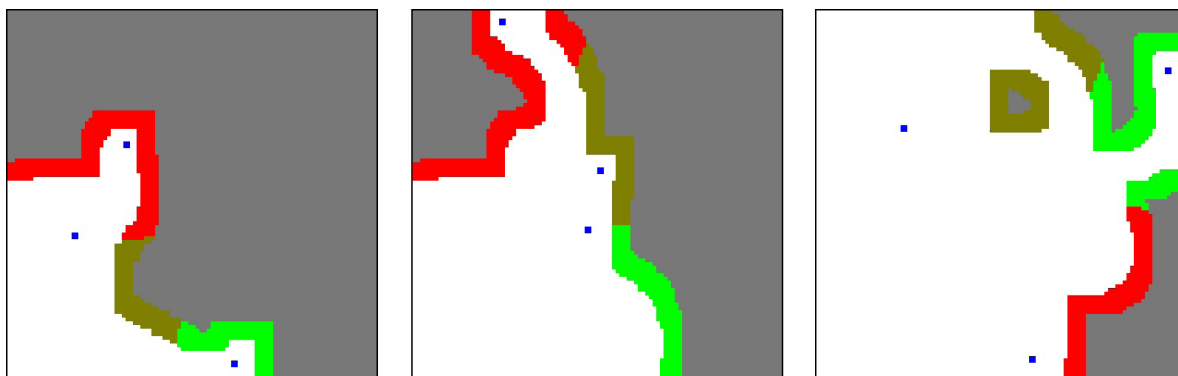


Рис. 1. Кластеризация граничной области. Серым цветом выделена неисследованная область, синие прямоугольники обозначают ТС, разными цветами выделены рассчитанные кластеры граничной области, белым цветом обозначена часть исследованной области, не являющаяся граничной

Подбор наилучших пар ТС-кластер. После кластеризации ГО необходимо сопоставить каждому из активных ТС некоторый кластер. Это выполняется с помощью Венгерского алгоритма [16]. Данный алгоритм выполняет поиск паросочетания минимального веса в полном двудольном графе, одной доле которого соответствуют ТС из группы, а другой кластеры. В качестве веса ребра каждой паре ТС-кластер сопоставляется длина пути из текущего положения ТС в ближайшую клетку кластера. Если пути не существует, то вес считается бесконечно большим.

Выбор целей для ТС. Далее для каждого ТС необходимо выбрать точку назначения из сопоставленного кластера граничной области. Для этого каждой клетке кластера сопоставляется числовое значение, определяющее выгодность ее выбора, с помощью функции приоритета.

Вид этой функции сильно влияет на работу алгоритма, так как определяет выбор точки назначения. Проведено сравнение работы алгоритма исследования при использовании различных функций (параметры сравнения приведены ниже) (рис. 2). В результате была выбрана функция, учитывающая следующие характеристики клеток:

- длина пути рассматриваемого ТС в данную клетку;
- удаленность клетки от текущих целей других ТС (увеличение данного параметра позволяет более эффективно распределить ТС по неисследованной области);
- удаленность клетки от ближайшей неисследованной клетки (чем меньше значение данного параметра, тем больше новой информации о территории будет получено при прибытии в клетку).

Таким образом, в качестве точки назначения выбирается клетка кластера, в которой принимает минимальное значение функция

$$f = \frac{T}{D \cdot C^2}.$$

Здесь T – расстояние по прямой до ближайшей из целей остальных ТС, D – расстояние по прямой до ближайшей неисследованной клетки, C – длина пути ТС из текущего положения в эту клетку.

Оценка асимптотической сложности алгоритма

Пусть N – общее количество свободных клеток, G – количество граничных клеток, K – количество ТС (равное количеству кластеров), I – количество итераций алгоритма K-means, T – размер шаблона поля зрения.

Оценка вычислительной сложности частей алгоритма составляет:

- расчет длин путей из всех ТС в свободные клетки карты – зависит от выбора алгоритма поиска пути. Для использованного алгоритма составляет $O(N^2 \cdot K)$;
- выделение ГО: $O(N \cdot T)$;
- кластеризация ГО: $O(G \cdot K \cdot I)$;
- нахождение паросочетания минимального веса между набором кластеров и набором ТС: $O(K^3)$;
- выбор точек назначения для каждого ТС из соответствующего кластера: $O(G)$;
- вычисление пути для каждого ТС в выбранную точку назначения: $O(N \cdot K)$.

Тогда оценка вычислительной сложности алгоритма составляет $O(N^2 K + NT + GKI + K^3 + G + NK) = O(N^2 K + NT + GKI + K^3)$. Так как $G < N$, $K < N$, $T < N$ и $I < N$, то значение общей оценки сложности алгоритма равно $O(N^2 K)$.

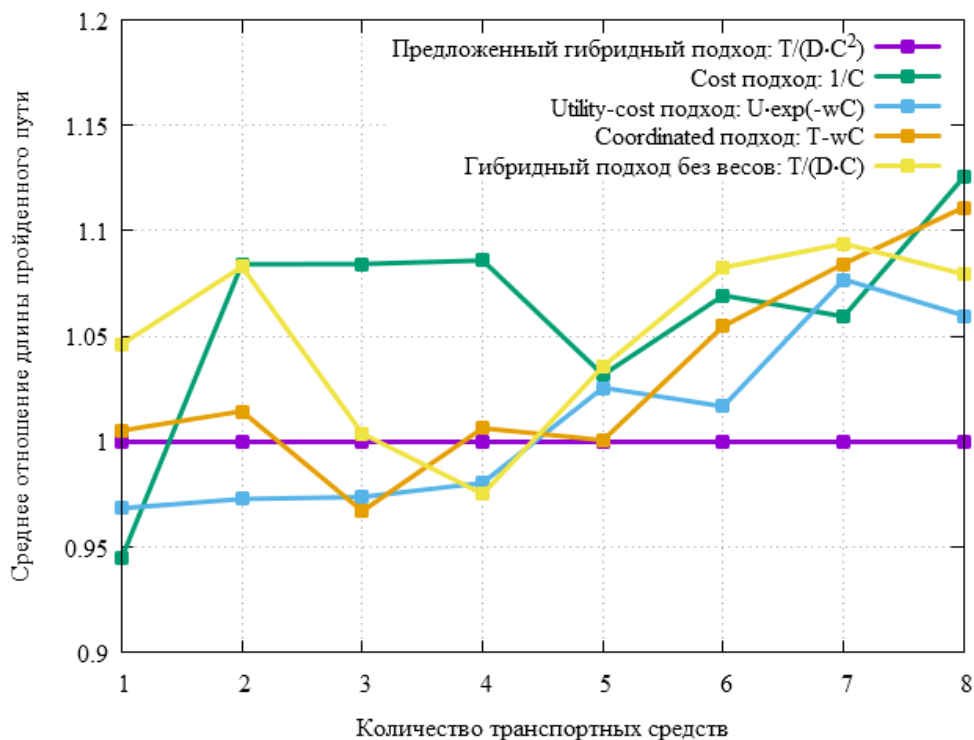


Рис. 2. Сравнение суммарной длины пути при использовании различных функций приоритета. Приведены значения среднего отношения суммарной длины пути при использовании существующих подходов к значению для предложенного подхода

Реализация подхода и ее тестирование

Описанный алгоритм реализован в виде программной библиотеки. Далее описаны особенности реализации, результаты тестирования ее характеристик и сравнения с аналогами.

Представление карты препятствий. В реализации разработанного алгоритма положения ТС на карте препятствий описываются несколькими координатами:

- двумерная вещественная координата;
- текущий угол поворота;
- направление движения (вперед или назад).

В данном алгоритме карта препятствий представлена в виде сетки. Клетки данной сетки также являются четырехмерными. При этом используется гибридная модель представления точек карты [17]. Каждая точка карты с вещественными координатами привязана к некоторой клетке карты препятствий (которая, в свою очередь, имеет целочисленные координаты). Для вычислений используются вещественные координаты точек, которые хранятся (при необходимости) в связи с соответствующей клеткой. Такой способ представления точек карты позволяет ограничить количество используемой памяти, при этом сохранив возможность вычислений в вещественных числах.

Требования к реализации алгоритма. Требования к реализации обусловлены следующими причинами. Во время работы реализации изменяются текущие данные карты препятствий и текущее положение ТС. Кроме того, необходима возможность работы с картой различных размеров, с различными параметрами ТС (размерами, формой и параметрами кинематической модели) и с различными шаблонами поля зрения.

Система представлена в виде библиотеки с возможностью реализации специфической для конкретной задачи функциональности сторонними разработчиками. Возможна реализа-

ция алгоритма, рассчитывающего пересечение модели ТС с препятствиями (в зависимости от геометрических параметров ТС), алгоритма, рассчитывающего движение ТС (в зависимости от кинематической модели), и алгоритма, рассчитывающего предполагаемое при расчетах поле обзора ТС. При тестировании реализации использованы модель прямоугольного автомобиля с двумя колесными осями и поле обзора квадратной формы.

Важна производительность реализации. Этим обусловлен выбор в качестве языка программирования C++. Функциональные и нефункциональные требования приведены в табл. 1, 2.

Таблица 1

Функциональные требования

№	Описание
1.	Программа должна поддерживать размеры карты препятствий до 1000×1000 клеток
2.	Программа должна поддерживать любые размеры АТС, не превышающие размера карты
3.	Входные данные: текущие положения АТС, карта препятствий, размеры карты препятствий и размеры АТС должны подаваться программе в виде входных параметров
4.	Результатом работы программы должен быть набор проезжаемых путей для АТС

Таблица 2

Нефункциональные требования

№	Описание
1.	Программа должна работать с предоставленной извне кинематической моделью АТС
2.	Программа должна работать с предоставленной извне функцией проверки столкновения АТС с препятствиями
3.	Программа должна работать с предоставленной извне функцией вычисления предполагаемого при расчетах поля обзора ТС
4.	Программа должна быть реализована с использованием языка программирования C++
5.	Среднее время подсчета не должно превышать 0,3 сек. на карте размера 100×100 *
6.	Максимальное время подсчета не должно превышать 1 секунды на карте размера 100×100 .

* Время указано для ПК с параметрами, описанными ниже в разделе Тестирование.

Особенности реализации. Реализация представленного алгоритма поддерживает два режима.

1. Точный. Все расчеты путей выполняются с использованием четырехмерных координат ТС.

2. Ускоренный. Расчет путей в граничные клетки использует трехмерные координаты состояний (учитывающих координаты ТС на карте и текущий угол поворота). Если в этом случае нет достижимых клеток, выполняется расчет с использованием четырехмерных координат.

Использовалась реализация алгоритма K-means из библиотеки OpenCV [18].

В реализации используется параллельное исполнение наиболее трудоемких составных частей алгоритма. При этом расчеты во многих составных частях не зависят друг от друга ни по данным, ни по управлению. Поэтому если количество ядер процессора не меньше количества ТС, то асимптотическая сложность алгоритма сокращается до $O(N^2)$ (см. выше).

Тестирование. Для замера количественных параметров реализации использовалась ЭВМ на базе процессора Intel Xeon X5660 с тактовой частотой 2,8 ГГц и объемом оперативной памяти 8 Гб. Использовались карты препятствий различного размера до 100×100 клеток.

Параметры ТС были взяты с прототипа реального автомобиля их значения отражены в табл. 3.

Таблица 3

Параметры ТС, использованные при тестировании

Параметр	Значение
Расстояние между осями колес автомобиля	2,5 ед.*
Расстояние от задней оси колес до центра автомобиля	1,25 ед.
Скорость движения автомобиля	1,9 ед./с
Максимальный угол поворота колеса	0,25 рад.
Размеры ТС (длина \times ширина)	$4 \times 1,8$ ед.

* 1 ед. соответствует ширине клетки карты препятствий.

При тестировании эффективности реализации выполнялись измерения двух наиболее важных количественных характеристик: суммарная длина пути, пройденного группой ТС при исследовании, и среднее время выполнения итерации. Анализ эффективности работы алгоритма имеет следующие особенности.

Во-первых, время работы итерации увеличивается при увеличении количества ТС. При этом чем больше ТС участвуют в исследовании, тем меньше итераций алгоритма необходимо совершить. Таким образом, сокращается длина пути, пройденного каждым ТС из группы.

Во-вторых, если в момент начала выполнения итерации алгоритма всем ТС из группы уже назначен актуальный путь (т. е. он является полезным для исследования и ТС еще не достигло точки назначения), то существенная часть расчетов не выполняется. Поэтому среднее время выполнения итерации алгоритма сокращается в зависимости от количества таких итераций. Эта особенность также сильно сокращает среднее время работы итерации алгоритма при уменьшении количества ТС, так как при этом увеличивается вероятность возникновения вышеописанной ситуации на каждой итерации алгоритма.

Тестирование производительности выполнялось на двадцати картах различного размера с различным количеством ТС. Данные о среднем времени выполнения итерации алгоритма приведены в табл. 4.

Сравнение с аналогами. Выполнено сравнение характеристик разработанного алгоритма с двумя близкими аналогами.

1. Алгоритм [13], использующий похожую функцию выбора точек назначения, но не использующий кластеризацию.

2. Алгоритм [14], использующий кластеризацию неисследованной области и более простую функцию выбора точек назначения.

Таблица 4

Среднее время выполнения итерации алгоритма, мс

Размер карты препятствий	Количество ТС		
	1	4	7
40 × 40	7	55	74
60 × 60	15	74	198
100 × 100	33	110	223

По вышеперечисленным аналогам существуют академические публикации, но нам не известны общедоступные реализации данных алгоритмов. Поэтому для проведения сравнения реализованы версии этих алгоритмов, результаты приведены на рис. 3 и 4.

На рис. 5 изображен процесс исследования тестовой карты с помощью реализованного алгоритма.

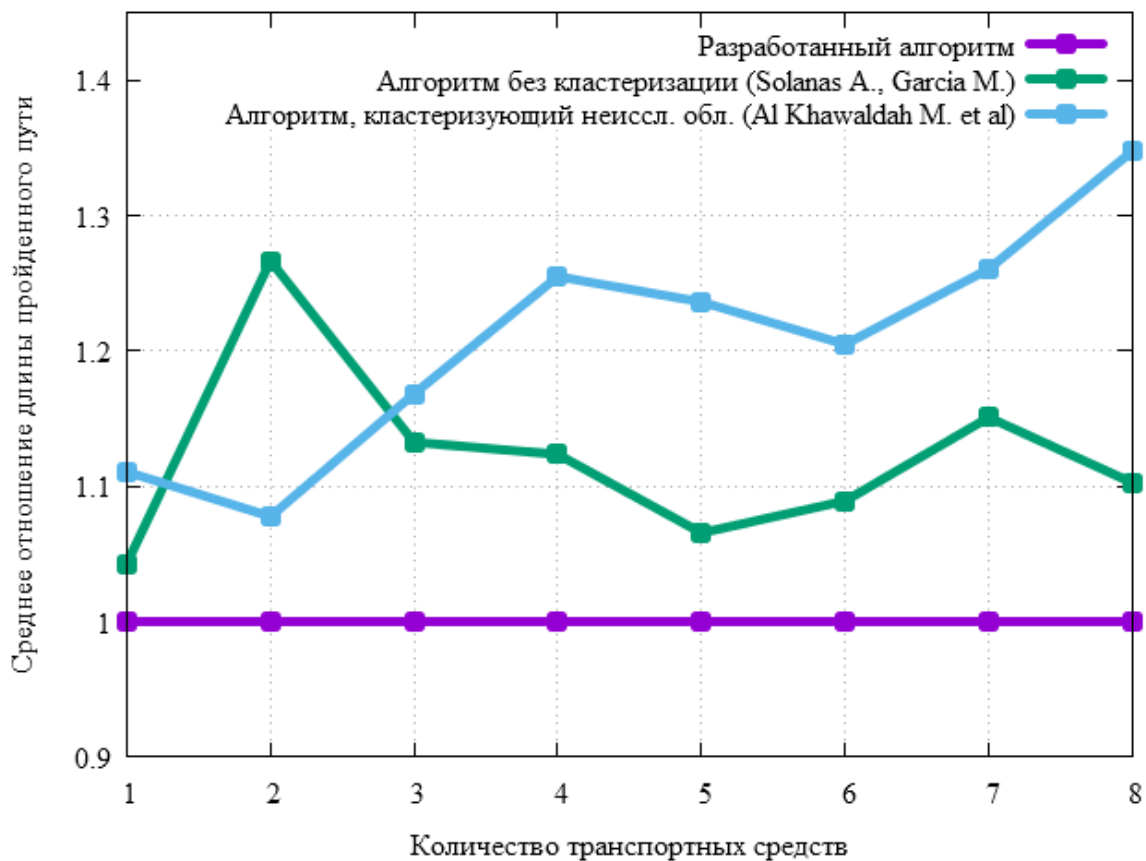


Рис. 3. Сравнение среднего отношения суммарной длины пройденного пути. Приведены значения среднего отношения суммарной длины пути реализаций аналогов к значению для реализации разработанного алгоритма

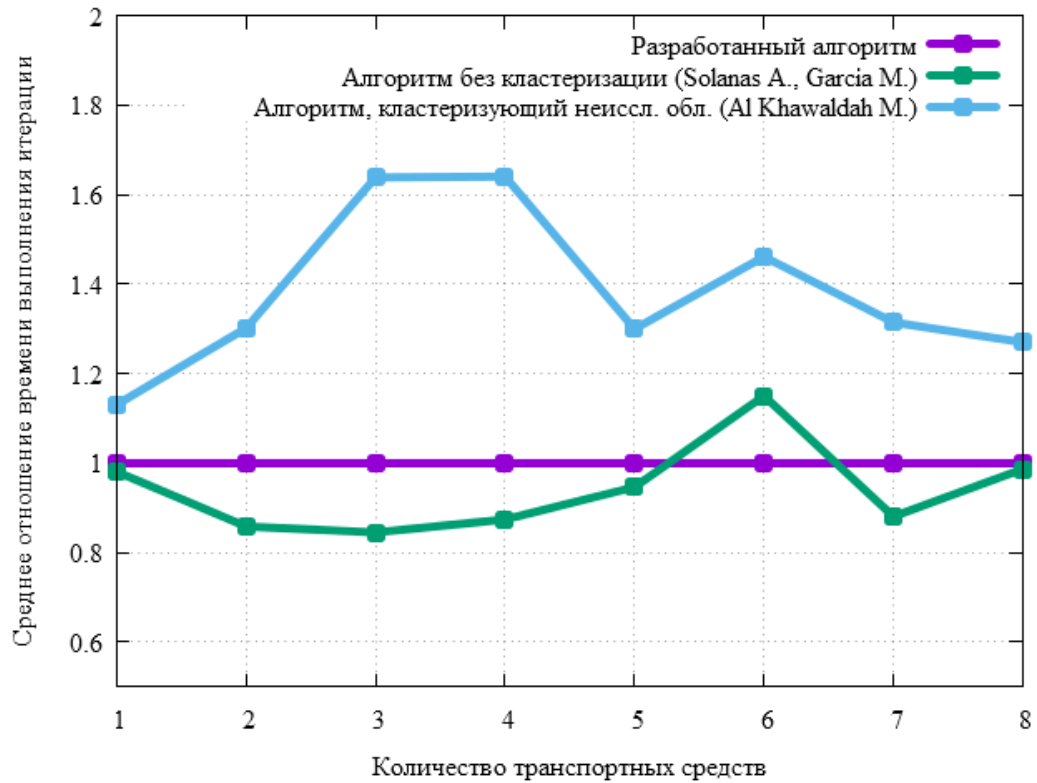


Рис. 4. Сравнение времени выполнения итерации. Приведены значения среднего времени выполнения итерации реализаций аналогов к значению для реализации разработанного алгоритма

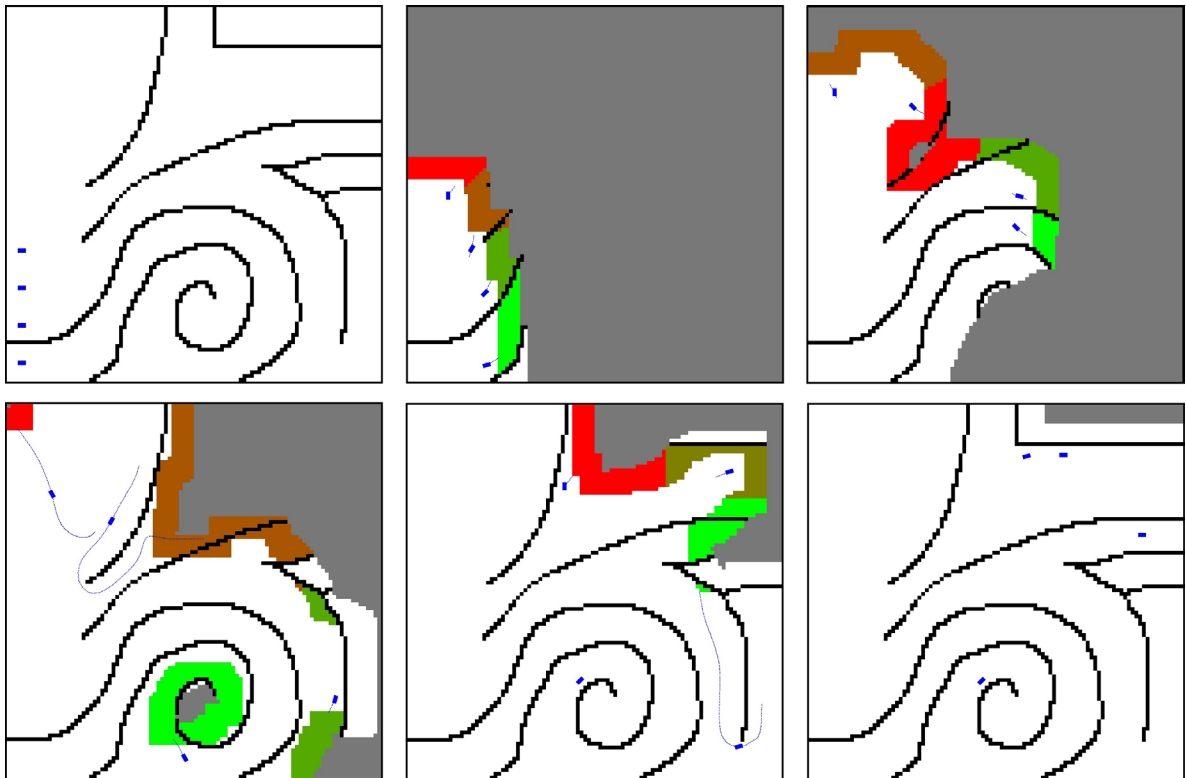


Рис. 5. Исследование местности с помощью четырех ТС. Черным цветом обозначены исследованные препятствия, серым – неисследованная область. Прямоугольники синего цвета обозначают ТС (размер прямоугольника соответствует размеру ТС, использованному при расчетах), линии синего цвета – текущие пути ТС. Разными цветами выделены рассчитанные кластеры граничной области, белым цветом обозначена часть исследованной области, не являющаяся граничной

Заключение

В работе представлен гранично-ориентированный алгоритм исследования заранее неизвестной территории с помощью группы АТС. Алгоритм основан на новом методе выбора точек назначения для АТС. Данный метод использует кластеризацию граничной области – области карты, находящейся на границе исследованной части. Каждому АТС сопоставляется некоторый кластер. Точка назначения для АТС может быть выбрана только из соответствующего ему кластера.

Использование кластеризации граничной области позволяет распределить неисследованную область между АТС. Это позволяет уменьшить суммарную длину пути группы АТС, пройденного при исследовании. В отличие от алгоритмов, использующих кластеризацию всей неисследованной области, кластеры граничной области всегда достижимы группой АТС. Кроме того, граничная область часто содержит меньше клеток, чем неисследованная область, поэтому кластеризация выполняется быстрее.

Выполнено сравнение разработанного алгоритма с двумя близкими аналогами: алгоритмом, не использующим кластеризацию и использующим кластеризацию неисследованной области. Сравнение на наборе тестовых карт показало, что суммарная длина пути АТС при использовании представленного алгоритма меньше, чем у аналогов.

Список литературы

1. *Gabrielly Y., Rimon E.* Spanning-tree based coverage of continuous areas by a mobile robot // *Annals of Mathematics and Artificial Intelligence*. 2001. No. 31. P. 77–98.
2. *Hazon N., Kaminka G.* On redundancy, efficiency and robustness in coverage for multi-robot // *Robot Autonomous System*. 2008. No. 56. P. 1102–1114.
3. *Murphy L., Newman P.* Using incomplete online metric maps for topological exploration with the gap navigation tree // *IEEE International Conference on Robotics and Automation*, 2008. P. 2717–2722.
4. *Santosh D., Achar S., Jawahar C. V.* Autonomous image-based exploration for mobile robot navigation // *IEEE International Conference on Robotics and Automation*, 2008. P. 47–60.
5. *Andries M., Charpillet F.* Multi-robot exploration of unknown environments with identification of exploration completion and post-exploration rendezvous using ant algorithms // *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2013. P. 5571–5578.
6. *Liu T. M., Lyons D. M.* Leveraging Area Bounds Information for Autonomous Multi-Robot Exploration.
7. *Senthilkumar K., Bharadwaj K.* Multi-robot terrain coverage by constructing multiple spanning trees simultaneously // *International Journal of Robotics and Automation*. 2010. Vol. 3. No. 25. P. 195–203.
8. *Lau H.* Behavioural approach for multi-robot exploration // *Proc. of the 2003 Australasian Conference on Robotics and Automation*, 2003.
9. *Yamauchi B.* A frontier-based approach for autonomous exploration // *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 1997. P. 146–151.
10. *Gonzalez-Banos H. H., Latombe J. C.* Navigation strategies for exploring indoor environments // *The International Journal of Robotics Research*. 2002. Vol. 21, 10–11. P. 829–848.
11. *Marjovi A., Marques L.* Multi-robot topological exploration using olfactory cues // *Distributed Autonomous Robotic Systems*. 2013. P. 47–60.
12. *Burgard W., Moors M., Fox D., Simmons R., Thrun S.* Collaborative multi-robot exploration // *IEEE International Conference on Robotics and Automation*. 2000. Vol. 1. P. 476–481.

13. *Al Khawaldah M., Al-Khedher M., Al-Adwan I., Al Rawashdeh A.* An Autonomous Exploration Strategy for Cooperative Mobile Robots // *Journal of Software Engineering and Applications*. 2014. Vol. 7. No. 3. P. 142–149.
14. *Solanas A., Garcia M. A.* Coordinated multi-robot exploration through unsupervised clustering of unknown space // *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2004. Vol. 1. P. 717–721.
15. *Steinhaus H.* Sur la division des corp materiels en parties // *Bull. Acad. Polon. Sci.* 1956. Vol. 1. P. 801–804.
16. *Kuhn H. W.* The Hungarian method for the assignment problem // *Naval Research Logistics*. 1955. Vol. 2, 1–2. P. 83–97.
17. *Dolgov D., Thrun S., Montemerlo M., Diebel J.* Path Planning for Autonomous Vehicles in Unknown Semi-structured Environments // *The International Journal of Robotics Research*. 2010. No. 29. P. 485–501.
18. *Bradski G.* Clustering and Search in Multi-Dimensional Spaces // *Dr. Dobb's Journal of Software Tools*. 2000.

Материал поступил в редколлегию

D. E. Kuzakov¹, M. S. Diakov², M. M. Lavrentyev³

PATH PLANNING FOR MULTI-ROBOT EXPLORATION USING FRONTIER SPACE CLUSTERIZATION

¹ *Novosibirsk State University*

2 Pirogov Str., Novosibirsk, 630090, Russian Federation

² *SoftLab-NSK Co. Ltd*

1 Acad. Koptyug Ave., Novosibirsk, 630090, Russian Federation

³ *Institute of Automation and Electrometry SB RAS*

1 Acad. Koptyug Ave., Novosibirsk, 630090, Russian Federation

kuzd@sl.iae.nsk.su, diam@sl.iae.nsk.su, mmlavrentiev@gmail.com

In this paper, a path planning algorithm for multi-robot exploration is presented. It is developed for exploration in initially unknown areas. The algorithm is based on a novel method of choosing exploration targets. This method uses clusterization of frontier space – part of explored map space which is situated on its border with an unexplored part. Every robot is being associated with a cluster. Then the exploration target for the robot is chosen from associated cluster with a priority function. This function defines utility for choosing a map cell considering traverse cost, information gain and distance to other robots' targets.

Keywords: multi-robot exploration, frontier space clusterization, frontier-based algorithm.

References

1. Gabrielly Y., Rimon E. Spanning-tree based coverage of continuous areas by a mobile robot. *Annals of Mathematics and Artificial Intelligence*, 2001, no. 31, p. 77–98.
2. Hazon N., Kaminka G. On redundancy, efficiency and robustness in coverage for multi-robot. *Robot Autonomous System*, 2008, no. 56, p. 1102–1114.
3. Murphy L., Newman P. Using incomplete online metric maps for topological exploration with the gap navigation tree. *IEEE International Conference on Robotics and Automation*, 2008, p. 2717–2722.

4. Santosh D., Achar S., Jawahar C. V. Autonomous image-based exploration for mobile robot navigation. *IEEE International Conference on Robotics and Automation*, 2008, p. 47–60.
5. Andries M., Charpillet F. Multi-robot exploration of unknown environments with identification of exploration completion and post-exploration rendezvous using ant algorithms. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, p. 5571–5578.
6. Liu T. M., Lyons D. M. Leveraging Area Bounds Information for Autonomous Multi-Robot Exploration.
7. Senthilkumar K., Bharadwaj K. Multi-robot terrain coverage by constructing multiple spanning trees simultaneously. *International Journal of Robotics and Automation*, 2010, vol. 3, no. 25, p. 195–203.
8. Lau H. Behavioural approach for multi-robot exploration. *Proceedings of the 2003 Australasian Conference on Robotics and Automation*, 2003.
9. Yamauchi B. A frontier-based approach for autonomous exploration. *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 1997, p. 146–151.
10. Gonzalez-Banos H. H., Latombe J. C. Navigation strategies for exploring indoor environments. *The International Journal of Robotics Research*, 2002, vol. 21, 10–11, p. 829–848.
11. Marjovi A., Marques L. Multi-robot topological exploration using olfactory cues. *Distributed Autonomous Robotic Systems*, 2013, p. 47–60.
12. Burgard W., Moors M., Fox D., Simmons R., Thrun S. Collaborative multi-robot exploration. *IEEE International Conference on Robotics and Automation*, 2000, vol. 1, p. 476–481.
13. Al Khawaldah M., Al-Khedher M., Al-Adwan I., Al Rawashdeh A. An Autonomous Exploration Strategy for Cooperative Mobile Robots. *Journal of Software Engineering and Applications*, 2014, vol. 7, no. 3, p. 142–149.
14. Solanas A., Garcia M. A. Coordinated multi-robot exploration through unsupervised clustering of unknown space. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004, vol. 1, p. 717–721.
15. Steinhaus H. Sur la division des corp materiels en parties. *Bull. Acad. Polon. Sci*, 1956, vol. 1, p. 801–804.
16. Kuhn H. W. The Hungarian method for the assignment problem. *Naval Research Logistics*, 1955, vol. 2, 1–2, p. 83–97.
17. Dolgov D., Thrun S., Montemerlo M., Diebel J. Path Planning for Autonomous Vehicles in Unknown Semi-structured Environments. *The International Journal of Robotics Research*, 2010, no. 29, p. 485–501.
18. Bradski G. Clustering and Search in Multi-Dimensional Spaces. *Dr. Dobb's Journal of Software Tools*, 2000.