

УДК 004.932.2+004.932.72'1

Е.С. ЛевицкаяДонецкий национальный технический университет, г. Донецк
кафедра программного обеспечения интеллектуальных систем**ИССЛЕДОВАНИЕ АЛГОРИТМА ВОССТАНОВЛЕНИЯ
ГРАФОВ КОЛЛЕКТИВОМ АГЕНТОВ****Аннотация**

Левицкая Е.С. Исследование алгоритма восстановления графов коллективом агентов. Предложена модификация алгоритма восстановления графов коллективом агентов. Исследованы свойства алгоритма восстановления графа. Определена структура взаимодействия агентов. Выполнена оценка сложности предложенного алгоритма и базового.

Ключевые слова: восстановление графа, разметка графа, коллектив агентов, топологическая среда, сложность алгоритма.

Постановка проблемы. Одной из центральных и актуальных задач робототехники является задача распознавания неизвестной среды блуждающими роботами. Существует множество алгоритмов для ее решения. Центральным моментом при распознавании среды – роботы могут мешать друг другу или блокировать друг друга. Актуальными проблемами являются задачи встречи двух роботов, а также прохождение робота по чужому пути. Таким образом, возникает необходимость в разработке оптимальных алгоритмов для робототехники, которые могли бы иметь минимальную емкостную, временную и коммуникационную сложность. Для реализации этого нужно выполнить следующие этапы:

- предложить алгоритм;
- исследовать свойства алгоритма;
- определить результаты исследования.

Анализ литературы. Алгоритм восстановления графа одним агентом представлен в [1], [2], в котором задействован агент исследователь, агент экспериментатор. При нахождении неразмеченных ребер агент исследователь использует «камень», который оставляет на вершине, в которой он «увидел» неразмеченное ребро, а затем начинает возврат по пройденному пути до тех пор, пока не увидит «камень». Основной алгоритм восстановления графов представлен в [3], [4]. В алгоритме используется два агента исследователя (АИ) и агент экспериментатор(АЭ). АИ управляются при помощи команд АЭ, который в свою очередь обладает неограниченной памятью.

Цель статьи – разработать алгоритм восстановления графов коллективом агентов и исследовать его свойства.

Постановка задачи исследования. Необходимо разработать такой алгоритм функционирования коллектива агентов исследователей (АИ), которые передвигаются по топологической среде и передают полученную информацию агенту экспериментатору (АЭ), и АЭ, который по полученным данным восстанавливает граф.

В работе рассматриваются конечные, неориентированные графы без петель и кратных ребер. Понятия, которые не рассмотрены ниже, общеизвестны и с ними можно ознакомиться в [1,4]. Пусть $G = (V, E)$ – граф, где V – множество вершин, E – множество ребер (двухэлементных подмножеств (v, u) , где $v, u \in V$). Тройку $((v, u), u)$ будем называть инцидентором ребра (v, u) и вершины u . Множество таких троек обозначим I . Перешеек – ребро, которые соединяет вершины несовпадающих (чужих) цветов.

АИ могут окрашивать вершины графа, ребра и инциденторы, воспринимать эти отметки и на их основании осуществлять перемещение. Информацию о своих действиях они передают АЭ, который и строит представление исследуемого графа.

Агенты A и B передвигаются по графу из вершины v в вершину u по ребру (v, u) , могут изменять окраску вершин v, u , ребра (v, u) , инциденторов $((v, u), v)$, $((v, u), u)$. Находясь в вершине v , АИ воспринимает метки всех элементов окрестности $Q(v)$. И на основании этих меток определяет, по какому ребру будет перемещаться, и как будет окрашивать элементы графа. АЭ передает, принимает и идентифицирует сообщения, полученные от АИ, обладает конечной, неограниченно растущей внутренней памятью, в которой фиксируется результат функционирования АИ на каждом шаге и строится представление графа G , вначале неизвестного агентам, списками ребер и вершин.

На начальном этапе все неразмеченные вершины имеют цвет 1. АИ помещаются в произвольные и неизвестные им самим вершины графа и сразу же помечают вершины в цвет 2, и передают АЭ следующую информацию: («нашел»). Через конечное число шагов АИ закончит разметку вершин, а АЭ восстановит неизвестный граф до изоморфизма, т.е. распознает граф.

Решение задач и результаты исследований. Возможности агентов: АИ: ход вперед, ход назад, использует 2 краски. АЭ: имеет неограниченную память, принимает от АИ сообщение, в случае необходимости запрашивает координаты АИ для распознавания перешейков. Спутник определяет местоположение АИ, отправляет координаты экспериментатору.

Краски агентам необходимы для того, чтобы агенты распознавали свой путь. Используя одну краску, агенты при попадании в любую вершину будут

считать её не пройденной, т.е. запрос на координаты агентов будет происходить на каждой вершине. Для того чтобы это избежать используем 2 краски: АИ1 размечает и ходит по пути 2 цвета, АИ2 размечает и ходит по пути 3 цвета.

Для решения поставленной задачи предлагается следующий алгоритм:

1. Помещаем агентов в произвольные вершины.
2. АИ1 и АИ2 размечает вершины цветом 2 и 3 соответственно, размечает цветом 2 и 3 их конечные инциденторы.
3. Передаем информацию («нашел»).
4. АЭ строит вершину, нумерует её, ставит в соответствие вершине координаты, полученные с помощью спутника.
5. АИ идет вглубь графа, пока есть вершины цвета 1. Иначе начинает движение назад, размечать конечные инциденторы, при этом начинается поиск перешейков.
6. В случае если перешеек найден, АИ отправляет информацию АЭ («нашел»). АЭ принимает сообщение, делает запрос на получение координат АИ, и соединяет ребром вершины: из которой пришел АИ в которую пришел, путем приведения соответствия координат.
7. Путь назад продолжается до тех пор, пока АИ не придет в начальную вершину. Начальной вершиной будет считаться вершина, в которой нет размеченных смежных вершин и инциденторов.
8. Если АИ1 пришел в начальную вершину, а АИ2 идет назад, то АИ1 отправляет сообщение об окончании работы.
9. Если АИ1 пришел в начальную вершину, а АИ2 идет вперед, то АИ1 перебрасывается в неразмеченную вершину с неразмеченной 2-окрестностью.
10. Если АИ2 пришел в начальную вершину, а АИ1 идет назад, то АИ2 отправляет сообщение об окончании работы.
11. Если АИ2 пришел в начальную вершину, а АИ1 идет вперед, то АИ2 перебрасывается в неразмеченную вершину с неразмеченной 2-окрестностью.
12. Поиск продолжается до тех пор, пока все вершины не будут размечены, т.е. у АИ не будет путей цвета 1 для передвижения.

Графическое представление части алгоритма представлено на рис. 1.

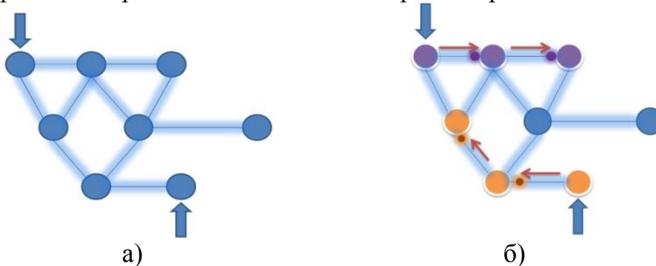


Рисунок 1 – Работа мобильных агентов: а) исходный граф, б) разметка агентов исследователей

Структура работы агентов исследователей, агента экспериментатора и запросы для определения местоположения представлены на рис. 2.

На рисунке 2 видно двустороннюю связь Спутник – АЭ, АИ – АЭ. АИ взаимодействует только с экспериментатором, следовательно, емкостная сложность АИ не увеличивается. АЭ в алгоритме с модификациями начинает взаимодействовать со спутник, что увеличивает коммуникационную сложность алгоритма. Результаты сравнений базового алгоритма и алгоритма с модификациями представлено в таблице 1.

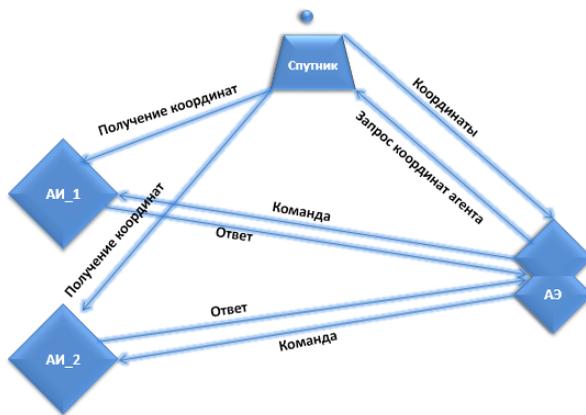


Рисунок 2 – Взаимодействие объектов

Таблица 1 – Сравнение сложности алгоритмов

Параметр \ Алгоритм	Временная сложность	Емкостная сложность	Коммуникационная сложность
Базовый алгоритм	$O(n^3)$	$O(n^2)$, 3 краски	$O(n)$
Алгоритм с модификациями	$O(n)$	$O(n)$, 2 краски	$O(n^2)$

Анализ полученных результатов, показывает увеличение значений коммуникационной сложности, и уменьшение значений временной сложности и емкостной. Такие результаты достигаются за счет следующих модификаций:

1. Выбор ребра и проход по нему осуществляется за единицу времени. В базовом алгоритме временная сложность $O(n^3)$ – это $O(n)$: прохождение АИ вперед* $O(n)$: прохождение АИ назад* $O(n)$: распознавание перешейков. В алгоритме с модификациями временная сложность $O(n)$ – это синхронное прохождение АИ вперед (т.е. за единицу времени АИ восстанавливают 2 вершины) и назад. При распознавании перешейков агенты не производят возвраты, за счет приведения в соответствие вершины и координат вершины.

2. Емкостная сложность определяется размером списков вершин цвета 1 и цвета 2. АИ используют 2 цвета, размечая 1 цветом вершину и инцидентор. Третий цвет не используется, так как АИ возвращается по размеченным конечным инциденторам.

3. Коммуникационная сложность увеличивается за счет двойного обмена сообщениями: сначала АИ: «нашел» – АЭ: «дальше» или АИ: «конец» – АЭ: «назад», а также АИ: «нашел» (перешеек) – АЭ: «запрос координат» – АЭ: «дальше».

Выводы. Исследован базовый алгоритм восстановления графа коллективом агентов. Предложена модификация для базового алгоритма. Исследованы свойства улучшенного алгоритма. Произведена оценка сложности базового алгоритма и алгоритма с модификациями. Результаты показали, что в разработанном алгоритме значительно уменьшена временная и емкостная сложность, за счет уменьшения количества возвратов при поиске перешейков на графе. Увеличена коммуникационная сложность, за счет запроса координат агентов исследователей для агента экспериментатора с целью распознать перешейки, что сильно не повлияло на скорость выполнения процесса восстановления графа, т.к. запросы координат выполняет АЭ, у которого неограниченная память.

Список литературы

1. Грунский И.С., Сапунов С.В., Детерминированная разметка вершин графа блуждающим по нему агентом, ПДМ. Приложение, 2012, С. 89–91.
2. Грунский И.С., Сапунов С.В., О самолокализации мобильного агента с использованием топологических свойств среды, ПДМ, 2011, С. 90–91.
3. Грунский И.С., Татаринев Е.А., Распознавание графа при помощи блуждающего по нему агента, ПДМ, 2009, С. 96–98.
4. Степкин А.В., Алгоритм распознавания конечных графов тремя агентами // Theoretical and applied aspects of cybernetics. Proceedings of the International Scientific Conference of Students and Young Scientists (Kyiv, February 21-25 2011), С. 185–188.
5. Ахо А. Построение и анализ вычислительных алгоритмов / А. Ахо, Дж. Хопкрофт, Дж. Ульман — М.: Мир, 1979. — 536 с.