УДК 519.7

Н.К. Шатохина (канд.техн.наук, доц.)[1],
П.А. Шатохин (канд.техн.наук, доц.)[2]
Государственный университет информатики и искусственного интеллекта[1]
nshatokh@rambler.ru
Донецкий национальный технический университет[2]
pshatokh@mail.ru

РАСПОЗНАВАНИЕ ГРАФА МОЗАИЧНОЙ СТРУКТУРЫ КОЛЛЕКТИВОМ АГЕНТОВ

Рассмотрена проблема анализа дискретных структур, представленных графом специального вида. В частности, рассмотрена задача описания структуры графа на основе информации, полученной при обходе его по границе. Описан алгоритм решения задачи, приведены оценки его временной и емкостной сложности.

автомат, агент, граф, алгоритм

Введение

В данной работе рассматривается тематика, связанная с автоматным анализом геометрических сред, графов, и других дискретных структур [1]. Одной из задач данной тематики, является задача описания структуры объекта на основе информации полученной при обходе графа по его границе. Подобные задачи возникают в робототехнике, молекулярной физике: построение модели карты среды, модели объекта [2].

Известен ряд работ (см. например [3]) посвященных задаче распознавания структуры объекта, в которых рассматривался случай решения подобных задач с использованием одного автомата (далее агента).

В настоящей работе предлагается алгоритм решения задачи распознавания с использованием двух агентов. В качестве модели объекта используется неориентированный граф. Первый агент-исследователь (АИ) выполняет перемещения по исходному графу и передачу информации второму агенту-экспериментатору (АЭ). Второй агент по полученной информации описывает структуру графа.

Основные определения и обозначения

В данной работе рассматривается неориентированный граф G без дыр, петель и кратных ребер. Для графа G(V,E), где V – множество вершин, E $=\{(v_i,v_j)|\ (v_i,v_j)\in V\}$ – множество ребер, заданы такие ограничения, описывающие его мозаичную структуру:

1) для всех $v\in V$ выполняется соотношение $1\le \deg(v)\le 6$;
2) для всех внутренних вершин выполняется $\deg(v)=6$ и существуют шесть маршрутов длины 3 из v в v, включающие всевозможные пары соседних ребер;
3) длина всех ребер $e=(v_1,v_2)\in E$ постоянна, $|e| = \text{const}$.

Граф называется сильно связным, если для любых двух v, w его вершин существует маршрут $(v,v_1),(v_1,v_2),\ldots,(v_{n-1},w)$, связывающий эти вершины.

Обозначим множество инцидентных ребер вершины v через $E(v)=\{e_1, \ldots\}$, $|E(v)| \le 6$. На E(v) определим функцию q: $E(v)\rightarrow H=\{a,b,c\text{-}a,\text{-}b,\text{-}c\}$. Множество H описывает метки инцидентных ребер вершины v, расставляя их смежным ребрам в порядке поворота на угол 60° против часовой стрелки относительно некоторого случайно взятого ребра. Знак «-» перед a означает, что ребро является диаметрально противоположным ребру a. Можно также интерпретировать H как множество $\{x, x^{+1}, x^{+2}, x^{+3}, x^{+4}, x^{+5}\}$, степень букв обозначает количество поворотов против часовой стрелки на угол 60° относительно ребра x. В этом случае имеем следующее: $x^{+3}=\bar{x}$, $x^{+4}=\bar{x}^{+1}$, $x^{+5}=\bar{x}^{+2}$. Как видим, функцию q можно интерпретировать как направление, так и угол поворота относительно некоторого ребра.

Утверждение 1. Любому маршруту $M=(v_1,v_2),(v_2,v_3),\ldots,(v_n, v_{n+1})$ взаимно однозначно соответствует строка направлений $S(v_1,v_2, \ldots,v_{n+1})=s_1s_2\ldots s_n$.

Доказательство. Утверждение следует из определения функции q , причем $s_1 s_2 \ldots s_n= q(v_1,v_2)\, q(v_2,v_3) \ldots q(v_n, v_{n+1})$.

Далее будем рассматривать маршруты по границам графа, в которых порядок просмотра ребер будет производиться против часовой стрелки.

Маршрут называется выпуклым, если для любых двух последовательных векторов $x^{li}\, x^{li+1}$ выполняется условие $0\le l_{i+1}-l_i\le 2$.

Нетрудно сообразить, что разность $l_{i+1}-l_i$ оценивает, какое по счету из H взято направление относительно x_i^{li} в качестве следующего вектора, или сколько поворотов на 60° составляет внешний угол между двумя этими векторами.

Граф называется выпуклым, если маршрут по граничным вершинам является выпуклым.

Маршрут называется вогнутым, если найдется хотя бы одна пара подряд следующих свободных векторов $x^{li}x^{li+1}$, для которых выполняется условие $l_{i+1}-l_i\ge 4$, а граф соответственно – вогнутым. Вершина, инцидентная этим ребрам, называется вершиной перегиба.

Базовые мозаики

Рассматриваемые в данной работе графы имеют мозаичную структуру, состоящую из правильных треугольников. Рассмотрим частные случаи таких графов, которые будем называть базовыми.

Равносторонний треугольник, это простейший случай мозаики. Маршруту по граничным ребрам мозаики в направлении против часовой стрелки соответствует строка направлений $S=x\, x^{+2}\, x^{+4}$, а по часовой стрелке – $S= x\, x^{+4}\, x^{+2}$. В общем случае – $S(k)=kx\ kx^{+2}\ kx^{+4}$ или $S(k)=kx\ kx^{+4}kx^{+2}$ соответственно, где параметр k – натуральное число равное количеству ребер, расположенных вдоль одной стороны треугольника.

Пусть вдоль стороны треугольника располагается l вершин (l-1 ребро), тогда общее количество вершин в мозаике выражается как $n=l(l+1)/2$, количество внутренних вершин как $m=l(l+1)/2 - 3(l - 1)$, и периметр p (количество шагов обхода по границе) оценивается как $3n \geq p \geq 3(\sqrt{1+8n}-3)/2$.

Параллелограмм – это граф, состоящий из двух равносторонних треугольников, имеющих общую сторону. Если первое и второе ребро маршрута составляют тупой угол, то маршруту по граничным ребрам мозаики в направлении против часовой стрелки соответствует строка направлений $S=xx^{+1}x^{+3}x^{+4}$, а по часовой стрелке - $S=x\ x^{+4}\ x^{+3}\ x^{+1}$. В общем случае имеем строки - $S(k_1, k_2)= k_1x\ k_2x^{+1}\ k_1x^{+3}\ k_2x^{+4}$ или $S(k_1, k_2)= k_1x\ k_2x^{+4}\ k_1x^{+3}\ k_2x^{+1}$, где параметры k_1, k_2 – количество ребер вдоль сторон мозаики.

Если же первое и второе ребро маршрута составляют острый угол, то маршруту против часовой стрелки соответствует строка направлений $S=xx^{+2}x^{+3}x^{+5}$ (для общего случая – строка $S(k_1, k_2)= k_1x\ k_2x^{+2}k_1x^{+3}k_2x^{+5}$), а для маршрута по часовой стрелке - $S=xx^{+5}x^{+3}x^{+2}$ (в общем случае – строка $S(k_1, k_2)= k_1x\ k_2x^{+5}k_1x^{+3}k_2x^{+2}$).

Параллелограмм, вдоль сторон которого располагается l_1 и l_2 вершин, имеет общее количество вершин $n=\begin{cases} l_1 \cdot l_2, \textit{если}\ l_1 \neq 0, \cdot\ l_2 \neq 0; \\ l_1, \textit{если}\ l_2 = 0; \end{cases}$, количество внутренних вершин - $m=\begin{cases} l_1 \cdot l_2 - 2(l_1+l_2-2), \textit{если}\ l_1 \neq 0, \cdot\ l_2 \neq 0; \\ 0, \textit{если}\ l_2 = 0; \end{cases}$, количество шагов обхода p по границе оценивается как $4(\sqrt{n}) \geq p \leq 2 \cdot n$

Трапеция состоит из трех равносторонних треугольников, имеющих общую вершину и два общих ребра. Строка $S=x\ x^{+1}x^{+2}x^{+4}$ описывает маршрут против часовой стрелки по границе мозаики. В общем случае этот маршрут описывает строка $S(k_1, k_2)= k_1x\ k_2x^{+1}\ k_1x^{+2}\ (k_1+ k_2)x^{+4}$, где k_1 – количество ребер вдоль боковой стороны, k_2 – вдоль меньшего основания мозаики. Строка $S=x\ x^{+5}x^{+4}x^{+2}$ описывает маршрут по часовой стрелке по границе мозаики, соответственно строка $S(k_1, k_2)= k_1x\ (k_1+ k_2)\ x^{+5}\ k_1x^{+4}\ k_2x^{+2}$ – маршрут в общем случае.

Указанная для трапеции строка имеет место, когда последовательность ребер рассматривается, начиная с правой боковой стороны. Все возможные трапеции (их всего четыре по числу углов), порождаются циклической перестановкой приведенной последовательности, однако они выражаются комбинацией треугольника и параллелограмма

Трапеция с количеством вершин l_1на боковой стороне и l_2 вершин на верхней стороне имеет общее количество вершин - $n= l_1(l_1 +2l_2 -1)/2$, внутренних вершин – $m= l_1(l_1-7)/2 + l_1l_2 - 2l_1+5$ и количество шагов p при обходе по границе при условии, что $k_1=k_2$ или $k_2=0$, будет $5\left(\frac{\sqrt{1+24n}+1}{6}-1\right) \leq p \leq 2n$.

Шестиугольник представляет собой граф, состоящий из шести треугольников с одной общей вершиной и шестью общими сторонами.

Маршруту по границе мозаики против часовой стрелки соответствует строка $S= x\ x^{+1}\ x^{+2}\ x^{+3}x^{+4}x^{+5}$, по часовой стрелке - $S= x\ x^{+5}\ x^{+4}x^{+3}x^{+2}x^{+1}$. Для произвольного случая маршруты описываются строками $S(k)= k_1x\ k_2x^{+1}\ k_1x^{+2}\ k_4\ x^{+3}k_5x^{+4}k_6x^{+5}$ (против часовой стрелки) и $S(k)= k_1x\ kx^{+5}\ kx^{+4}\ kx^{+3}\ kx^{+2}\ kx^{+1}$ (по часовой стрелке), где $k_1=k_3$, $k_5=k_2$, $k_4=k_6$ или $k_1=k_4$, $k_5=k_3$, $k_2=k_6$.

Шестиугольник с равным количеством вершин на всех боковых сторонах l имеет общее количество вершин $n= 3l^2 -3l+1$, внутренних вершин $m=3l^2 -9l+7$ и количество шагов p при обходе по границе при условии, что $k_1=k_2=k_3$ или $k_2=0$, будет $\sqrt{12n-3}-3 \le p \le 2n$.

Введем обозначения для строк, порождаемых при обходе базовых структур в направлении против часовой стрелки: +T(k) – строка направлений, соответствующая треугольнику с длиной стороны k, $+Пт(v,k_1,k_2)$ – параллелограмму с первым тупым углом, $+По(v, k_1, k_2)$ - параллелограмму с первым острым углом, $+Тр(v, k_1, k_2)$ - трапеции, $+Ш(k_1,k_2,k_3,k_4 ,k_5,k_6)$ – шестиугольнику. Строки, порождаемые при обходе базовых структур в направлении по часовой стрелке, будем обозначать со знаком «-». Поскольку соответствие между базовыми структурами и строками взаимнооднозначно, то эти обозначения являются и описателями базовых и обратных базовых структур.

Нетрудно заметить, что маршруты по границам базовых фигур, построенные руководствуясь правилом «по часовой стрелке», являются выпуклыми, а правилом «против часовой стрелки» - вогнутыми.

Вогнутый маршрут для треугольника содержит одну точку перегиба, для параллелограмма, трапеции – две точки перегиба, а для шестиугольника – пять.

Свойства агентов

Установленный в некоторую вершину графа, АИ может пользоваться некоторой локальной информацией о вершине – ее степенью. Он может выполнять повороты на 60° по часовой и против часовой стрелки, а также может перемещаться из вершины в вершину по ребрам и определять направление своего перемещения.

АИ выполняет следующие действия. Первую встреченную граничную вершину АИ метит. При передвижении по границе он фиксирует строку букв, соответствующих выбранным направлениям и передает эту строку АЭ. Агент прекращает работу, как только встретит ранее отмеченную вершину.

Агент АЭ принимает на каждом шаге сообщения от АИ, и описывает представление графа в символьном виде – в виде формулы, отражающей состав исходного графа как комбинацию базовых структур.

Постановка задачи

Требуется разработать такие алгоритмы передвижения агента АИ по исходному графу G с передачей информации об этом, и восстановления графа агентом АЭ по полученным данным в виде формулы, что АИ за конечное число шагов независимо от начального размещения обойдет часть внутренних вершин G и все граничные вершины, пошагово передавая информацию, по которой АЭ через конечное число шагов опишет граф P, изоморфный исходному графу G.

Свойства системы перемещений.

Любое перемещение АИ по любому ребру (операцию перемещения) можно представить свободным вектором, имеющим начало - в некоторой вершине v, конец – в вершине w, а x= q(v,w) задает направление свободного вектора. Другими словами между свободными векторами и направлениями, определяемыми функцией q, имеет место взаимно однозначное соответствие, поэтому данный вектор будем обозначать буквой x.

В качестве нулевого вектора, будем рассматривать вектор, у которого начало и конец совпадают, тогда x+0=x.

Для векторов определены правила сложения и умножения на действительные числа. Учитывая специфику мозаики графа, будем применять сложение векторов по правилу треугольников, например: $x+x^{+2} =x^{+4}$.

Для каждого вектора x имеется обратный вектор $\bar{x}$, сумма которых равна нулевому вектору. В данном случае, например $x+x^{+3}= x+\bar{x}=0$.

Определим результат сложения вектора с самим собой как умножение на числовой коэффициент: x+x= 2 x.

Остальные варианты сложения векторов представлены в виде таблицы Кэли.

Таблица 1 – Таблица Кэли

	x	$\mathbf{x^{+1}}$	$\mathbf{x^{+2}}$	$\mathbf{x^{+3}}$	$\mathbf{x^{+4}}$	$\mathbf{x^{+5}}$
x	2 x	$x^{+1}x$	x^{+1}	0	x^{+5}	$x^{+5}x$
$\mathbf{x^{+1}}$	xx^{+1}	$2\,x^{+1}$	$x^{+2}x^{+1}$	x^{+2}	0	x
$\mathbf{x^{+2}}$	x^{+1}	$x^{+1}x^{+2}$	$2\,x^{+2}$	$x^{+3}x^{+2}$	x^{+3}	0
$\mathbf{x^{+3}}$	0	x^{+2}	$x^{+2}\,x^{+3}$	$2\,x^{+3}$	$x^{+4}\,x^{+3}$	x^{+4}
$\mathbf{x^{+4}}$	x^{+5}	0	x^{+3}	$x^{+3}x^{+4}$	$2\,x^{+4}$	$x^{+5}x^{+4}$
$\mathbf{x^{+5}}$	xx^{+5}	x	0	x^{+4}	$x^{+4}x^{+5}$	$2\,x^{+5}$

Как видим, данная операция удовлетворяет свойством перестановочности. Нетрудно показать, что выполняется ассоциативность сложения. А поскольку для каждого вектора x имеется обратный вектор x′, то введенная операция определяет коммутативную группу.

Важным свойством коммутативной группы является факт, что сумма любого числа элементов в коммутативной группе не зависит от их порядка.

Операцией перемещения из вершины v по ребру e= (v,w) будем называть пару (v,e), где $v \in V$, $e=(v,w) \in E$. Учитывая функцию q, имеем (v,e)=(v, q(e)), где q(e) описывает направление перемещения из вершины v. Таким образом, операция перемещения позволяет задать функцию f: $(E(v), q(v,w)) \rightarrow V$ так, что f(v, q (v,w))=w, где (v,w) ребро с меткой q(v,w).

Лабиринтом L=(G, t, q, f) называется граф G, на котором определена t – разметка вершин, q – разметка ребер и f – функция перемещения.

Утверждение 2. Любому замкнутому маршруту в графе мозаичной структуры соответствует линейно зависимая комбинация свободных векторов, т.е. равная 0.

Действительно любому маршруту соответствует последовательность свободных векторов, для которой начало следующего вектора, совпадает с концом предыдущего. Если совокупность этих векторов составляет замкнутый маршрут, то сумма их равна 0. Что требовалось доказать.

Следует заметить, что вышеописанному последовательному применению операции перемещения по ребрам графа соответствует последовательность свободных векторов, начало которых совпадает с последовательностью пройденных вершин и направлениями векторов, и как раннее установлено, заданными строкой направлений $S(v_1,v_2,v_3,\ldots,v_{n+1}) = s_1,s_2,\ldots,s_n$, каждый символ s_i которой определяется перестановками множества H.

Таким образом, между последовательностью применения операции перемещения, последовательностью свободных векторов, и строкой их направлений существует взаимно однозначное соответствие.

Представлением базовой структуры M в G назовет строку M(v,k, …), где v – некоторая вершина графа, а M(k, …) – описатель базовой структуры.

Граничную вершину v далее будем называть опорной вершиной базовой структуры.

Утверждение 3. Множества векторов, соответствующие базовым и обратным базовым структурам, линейно зависимы, т.е. их сумма равна нулю.

Утверждение 3 является следствием утверждения 2.

Значит, например $+T(v,k)=S(k)= kx\ kx^{+4}\ kx^{+2}=\{\varnothing\}$ соответствующая строка направлений пуста.

В связи с тем, что эти строки однозначно генерируются при обходе графов, далее их обозначения будем называть короче базовыми и обратными базовыми структурами.

Идентификатором базовой структуры M(k,…) называется подстрока idM(k,…), полученная из S(k,…) путем удаления последнего символа с учетом соответствующего коэффициента, т.е. $M(k,\ldots)=idM(k,\ldots)\ kx^d$, где d –

степень последнего символа в M(k). Через M(v,k,…) будем обозначать факт расположения идентификатора в строке маршрута.

Как и ранее будем обозначать знаками «+», «-» идентификаторы выпуклых и вогнутых базовых структур. Например, +idШ(v,k)= $kx\ kx^{+1}\ kx^{+2}\ kx^{+3}\ kx^{+4}$– идентификатор шестиугольника.

Введем операции сокращения на подстроках, используя определение идентификатора:

$$a\ +idM(v,k)\ b= a\ k\bar{\mathbf{x}}^{d}\ b$$
$$a\ -idM(v,k)\ b= a\ k\bar{\mathbf{x}}^{r}\ b.$$

В обоих случаях происходит сокращение строки букв. Действительно, добавив после +idM(v,k) строку $k\mathbf{x}^{d}k\bar{\mathbf{x}}^{d} = \varnothing$, и перегруппировав подстроки, получим:

$$a\ +idM(v,k)\ b =a\ (+idM(v,k)\ k\mathbf{x}^{d} =\varnothing)k\bar{\mathbf{x}}^{d}\ b = a\ k\bar{\mathbf{x}}^{d}\ b.$$

Аналогично и для -idM(v,k):

$$a\ -idM(v,k)\ b =a\ (-idM(v,k)\ k\mathbf{x}^{r} =\varnothing)k\bar{\mathbf{x}}^{r}\ b = a\ k\bar{\mathbf{x}}^{r}\ b.$$

Здесь k,r – степени последних символов в идентификаторах +idM(v,k), -idM(v,k) соответственно.

Назовем направленной гранью (v,kx^{a}) маршрут M= $(v_1,v_2), (v_2,v_3), \ldots, (v_{n-1}, v_n)$, состоящий из k одинаково направленных ребер: $q(v_i,v_{i+1})=x^{a}$ для всех ребер $(v_i,v_{i+1})\in M$.

Направленную грань (v, kx^{a}) назовем гранью отсечения графа G(V,E), если существует два подграфа $G_1(V_1,E_1)$, $G_2\ (V_2,E_2)$, удовлетворяющие следующим свойствам:

$V= V_1\cup V_2$, $V_1 \cap V_2 = \{v,w_1,w_2,\ldots,w_{k-1},v' \mid w_1=f(v),w_2=f(w_1),\ldots,w_{k-1}=f(w_{k-2}), v'=f(w_{k-1})\}$;

$E= E_1\cup E_2$, $E_1\cap E_2 = \{ (v, w_1), (w_1,w_2),\ldots,(w_{k-1},v') \}$; $q(v, w_1)= q(w_1,w_2)=\ldots=q(w_{k-1},v')=x^{a}$

$|E_1| \geq 3$, $|E_2| \geq 3$, $|V_1|\geq 3$, $V_2|\geq 3$.

где x^{a} – направление ребер от граничной вершины v к граничной вершине v′, $w_1,w_2,\ldots,w_{k-1}$ – внутренние вершины графа, k –их количество.

Направленную грань $(v,\ k_1\ x^{a1}\ k_2\ x^{a2}\ldots\ k_h\ x^{ah})$ назовем гранью присоединения к графу $G_1(V,E)$, графа $G_2\ (V_2,E_2)$, если выполняется следующие свойства:

$V_1\cap V_2 = \{v, w_1,w_2,\ldots,w_{k-1},v' \mid w_1=f(v),w_2=f(w_1),\ldots,w_{k-1}=f(w_{k-2}), v'=f(w_{k-1})\}$;

$E_1\cap E_2=\{(v, w_1), (w_1,w_2),\ldots,(w_{k-1},v') \}$; $q(v, w_1)= x^{a1}$, $q(w_1,w_2)= x^{a2},\ldots =q(w_{k-1},v')=x^{ah}$.

$|E_1| \geq 3$, $|E_2| \geq 3$, $|V_1|\geq 3$, $V_2|\geq 3$.

где $x^{a1}\ x^{a2}\ldots\ x^{ah}$ – направление ребер от граничной вершины v к граничной вершине v′, $E_2 \setminus E_1$, в том числе и $w_1,w_2,\ldots,w_{k-1}$ – несуществующие вершины графа, $k_1+k_2+\ldots+k_h$ –их количество.

В соответствии с операциями сокращения в первом случае в графе происходит отсечение базовой структуры (+M(v,k) принадлежит исходному графу), отсечение происходит вдоль направления kx^{+d}, начиная с граничной вершины v. Во втором случае происходит присоединение -M(v,k) так, что из вершины v далее зафиксируется перемещение по несуществующим граничным вершинам графа G (-M(v,k) несуществующая часть графа).

Утверждение 4. Операция присоединения поглощает точки перегиба.

Доказательство. Утверждение следует из определения обратных базовых структур, согласно которым их ребра образуют точки перегиба.

Этим утверждением воспользуемся для преобразования сильно связного подграфа произвольной структуры, имеющего точки перегиба, к выпуклому виду.

Утверждение 5. Опорными точками идентификаторов базовых структур являются граничные вершины графа.

Доказательство следует из определений граней присоединения и отсечения, в которых граничная вершина v в случае базовых структур и является опорной точкой.

Распознавание графа

Процесс распознавания состоит из двух алгоритмов. Первый алгоритм «Обход» описывает перемещение АИ по неизвестному графу G и генерацию информации для АЭ. Второй алгоритм «Восстановление», представляет собой анализ полученной информации, на основании которой создается символьное описание графа P, изоморфного исходному G.

Алгоритм «Обход» работы АИ

Предполагается, что агент случайным образом может быть помещен в любую вершину графа, поэтому алгоритм состоит из двух этапов. На первом этапе, если необходимо АИ выходит на границу, а на втором этапе АИ обходит граф против часовой стрелки по граничным ребрам до тех пор, пока не возвратится в вершину выхода на границу.

Алгоритм АИ выхода на границу. Агент, помещенный в произвольную вершину v, определяет ее степень, если это внутренняя вершина, т.е. deg(v)=6, то перемещается по диаметрально противоположному ребру относительно первого встреченного ребра до следующей вершины. Эту процедуру повторяет до тех пор, пока не найдет граничную вершину, выбирает первое граничное ребро путем поворота против часовой стрелки, которое помечает маркером.

Алгоритм обхода графа по граничным вершинам. Дальнейшее перемещение агента происходит по границе графа в направлении против часовой стрелки, начиная из найденной вершины и ребра. Агент перемещается к следующей вершине по выбранному ребру, фиксируя букву-направление своего перемещения. Далее АИ проверяет, является ли помеченным маркером выбранное ребро. Если оно не помечено, он пересылает букву АЭ и продолжает движение, в противном случае АИ завершает свою работу.

Алгоритм «Обход» состоит двух частей. Первая часть может отсутствовать.

Лемма 1. Если в мозаичном графе n вершин, то максимальный путь по внутренним вершинам графа в выбранном направлении не превосходит величины l=(n-1)/3.

Лемма 2. Любой маршрут по внутренним вершинам графа в выбранном направлении перемещения является незамкнутым.

Исходный граф G может иметь более сложную структуру, чем базовая структура, для которой выполняется следующее.

Лемма 3. Длина маршрута p по граничным вершинам G $\sqrt{12n-3}-3 \le p \le 3n$.

При подсчете временной сложности имеет место:

- для выхода на границу графа согласно лемме 1 требуется (n-1)/3 шага, т.е. O(n);
- согласно лемме 3 обход по границе не превосходит величины 3n шагов, значит оценивается как O(n) шагов.

Емкостная сложность E(n) отсутствует, т.к. агенту ничего не требуется сохранять.

Алгоритм «Восстановление» работы АЭ

В алгоритме анализируется строка направлений, сгенерированная и переданная АИ.

Вначале производится поиск точек перегиба. Если точки имеются, то определяется идентификатор обратной базовой структуры, содержащий очередную точку перегиба, в качестве второй вершины маршрута. Если находится обратная базовая структура, то производится преобразование строки, наращивание счетчиков внутренних Сч_вв и граничных Сч_гв вершин, присоединение данной структуры к графу, что фиксируется добавлением в формулу со знаком «-» соответствующего фрагмента. В противном случае производится поиск идентификаторов базовых структур, расположенных до точки перегиба, пока не поглотится эта точка.

Далее вновь просматривается сначала строка для поиска идентификаторов базовых структур. Каждый раз, когда находится идентификатор, производится удаление из строки соответствующей подстроки, уменьшение значения Сч_вв и Сч_гв, отделение данной структуры от графа, и добавление в формулу со знаком «+» соответствующего фрагмента.

В алгоритме распознавания осуществляется два просмотра исходной строки.

Если имеется хотя бы одна точка перегиба, то в граф добавляются соответствующее базовой структуре количество вершин и устанавливается неявная нумерация u: $V(G \cup G') \rightarrow V(H \cup G')$, где G' граф включенной части. Нумерация u(v)=Сч_вв или u(v)=Сч_гв, для всех $v \in G'$. Эта нумерация является биекцией, т.к. добавление базовых структур выполняется взаимно однозначно в соответствии с их идентификаторами.

При выполнении операции отделения в графе H создается неявная нумерация d: $V(G') \to V(H)$ вершин графа G в вершины графа H, тех вершин, которые содержатся в отделяемой базовой структуре. При этом равенство $d(v)= t$, где t=Сч_вв или t=Сч_гв. Эта нумерация также является биекцией, т.к. удаление базовых структур выполняется взаимно однозначно в соответствии с их идентификаторами. Тем самым доказана следующая теорема.

Теорема 1. Выполняя алгоритмы распознавания, агенты распознают любой граф G с точностью до изоморфизма.

При подсчете временной сложности алгоритма работы АЭ учитывается, что распознавание структуры графа выполняется за счет двукратного просмотра исходной строки, длина которой не может превосходить 2n букв, т.е. временная сложность оценивается следующим образом. Процесс присоединения базовых структур выполняется не более чем возможное количество точек перегиба, а это не более чем за n/2 шагов. Процесс отсечения базовых структур выполняется не более чем возможное количество углов в графе, т.е. не более, чем за n/2 шагов. Второй просмотр строки выполняется не более чем за 2n шагов. Следовательно, суммарная временная сложность оценивается как $T(n)=O(n)$.

Таким образом, совместная работа АИ и АЭ оценивается как O(n).

Емкостная сложность E(n) определяется сложностью списков строки направлений S, множества граничных вершин $V' \subset V$, и строки формул F, сложность которых определяется величинами O(n), O(n), O(n).

В связи с ограничениями на объем подробные описания алгоритмов опущены.

Выводы

В работе предложен алгоритм восстановления структуры графа в виде формулы, отражающей состав исходного графа как комбинацию базовых структур. Далее предполагается рассмотреть графы, состоящие из сильно связных подграфов, соединенных мостами, перешейками и графы с дырами.

Список литературы

1. Капитонова Ю.В. Математическая теория проектирования вычислительных систем / Ю.В.Капитонова, А.А.Летичевский.-М.:Наука,1998.-296 с.
2. Dudek G., Jenkin V. Computational Principles of Mobile Robotics.-Cambridge University Press, Cambridge, 2000.-280p.
3. Кудрявцев В.Б. О поведении автоматов в лабиринтах / В.Б. Кудрявцев, Ш. Ушчумлич, Г. Килибарда // Дискретная математика.-1992.-Т.4, Вып.3.- С.3-28.

Надійшла до редакції 10.10.2010 *Рецензент: д-р техн.наук, проф. Скобцов Ю.А.*

Н.К. Шатохіна[1], П.А. Шатохін[2]

Державний університет інформатики і штучного інтелекту[1], Донецький національний технічний університет[2]

Розпізнавання графа мозаїчної структури колективом агентів. Розглянуто проблему аналізу дискретних структур, представлених графом спеціального виду. Зокрема, розглянуто задачу опису структури графа на основі інформації, отриманої при обході його по границі. Описано алгоритм розв'язання задачі, наведені оцінки його часової та ємнісної складності.
автомат, агент, граф, алгоритм

N.K. Shatokhina[1], P.A. Shatokhin[2]
State University of Informatics and Artificial Intelligence[1], Donetsk National Technical University[2]

Recognition of Mosaic Graph by a Collective of Agents. We consider the problem of analysis of discrete structures, represented as a graph of special kind. In particular, we consider the problem of describing the structure of the graph based on information obtained during its traversal of the boundary. The algorithm of solving the problem is described; we estimate its time and space complexity.
automaton, agent, graph, algorithm