УДК 519.174.5

АЛГОРИТМЫ КЛАСТЕРИЗАЦИИ ГРАФОВ

Г. Ш. Цициашвили, М. А. Осипова, А. С. Лосев

Дальневосточный федеральный университет,
Институт прикладной математики ДВО РАН

Поступила в редакцию 12.11.2014 г.

Аннотация. В настоящей работе строятся последовательные алгоритмы выделения компонент связности вершин в неориентированного графе и компонент сильной связности вершин в ориентированном графе. На каждом шаге работы алгоритма поступает информация об очередной вершине графа и о ребрах, соединяющих ее хотя бы с одной из вершин в каждой из выделенных ранее компонент связности. В ориентированном графе в процессе работы алгоритма на каждом шаге определяется отношение частичного порядка между компонентами сильной связности. В отличии от известных в литературе алгоритмов выделения компонент связности, основанных на поиске в глубину или в ширину, предложенный в данной работе алгоритм не возвращается повторно к уже обработанным вершинам в "дереве поиска" и не определяет для каждой вершины из "дерева поиска" всех смежных с ней.

Ключевые слова: отношение частичного порядка, компоненты связности, компоненты сильной связности, факторизация.

CLUSTERING ALGORITHM OF GRAPHS

G. Sh. Tsitsiashvili, M. A. Osipova, A. S. Losev

Abstract. In this paper sequantial algorithms of connectivity components separation in nondirected graphs and components of strong connectivity in directed graphs are constructed. On each algorithm step an information about next node and edges connected this node with earlier constructed clusters comes. In directed graph suggested algorithm defines a relation of partial order between clusters also. In comparison with known algorithms of connectivity components separation based on a search in a depth or in a width suggested algorithm does not return iteratively to processed nodes in "search tree" and does not define for each node from "search tree" all incident nodes.

Keywords: a relation of a partial order, a component of a connectivity, a component of a strong connectivity, a factorization.

ВВЕДЕНИЕ

Известные в литературе [1], [2] алгоритмы выделения компонент связности в неориентированных графах основаны на поиске в глубину или в ширину, а алгоритмы выделения компонент сильной связности в орграфах - на вариации алгоритма поиска в глубину (алгоритмы Косарайю, Тарьяна). Эти алгоритмы определяют все вершины, достижимые из выбранной s при построении "дерева поиска" , которое является компонентой связности, содержащей вершину s. Далее берется вершина, не попавшая в построенные ранее компоненты связности, и для нее строится "дерево поиска". Недостаток этих алгоритмов - это повторное возвращение к уже обработанным вершинам в "дереве поиска" и определение для каждой вершины из "дерева поиска" всех смежных с ней.

В настоящей работе для факторизации ориентированного и неориентированного графов предложены последовательные алгоритмы прохождения по всем вершинам графа и построение на каждом шаге классов эквивалентности, которые лишены перечисленных недостатков. А в случае ориентированного графа алгоритм последовательно устанавливает отношение частичного порядка между кластерами.

При работе алгоритма выделения компонент связности в неориентированном графе на шаге k поступает информация о связи k-ой вершины ребром с построенными на предыдущем шаге компонентами связности. Если такие компоненты связности найдены, то они объединяются в одну вместе
с k-ой вершиной. Иначе появляется новая компонента связности, состоящая из одной k-ой вершины. Алгоритм выделения компонент сильной связности в орграфе строился по аналогии с описанным. На k-ом шаге его работы поступает информация о наличии дуг между k-ой вершиной и построенными на $k-1$ шаге классами эквивалентности, между которыми установлено отношение частичного порядка. В зависимости от полученной информации происходит либо объединение классов эквивалентности между собой и с k-ой вершиной, либо появление нового класса, состоящего из одной k-ой вершины. При этом устанавливается отношение частичного порядка между образованными классами эквивалентности.

1. ФАКТОРИЗАЦИЯ ВЕРШИН НЕОРИЕНТИРОВАННОГО ГРАФА

Рассмотрим неориентированный граф с конечным множеством вершин $\{1, \dots, n\}$ и ребер. Зададим граф матрицей смежности $||d_{ij}||_{i,j=1}^n$ и положим $d_{ii} = 1,\ 1 \leqslant i \leqslant n$. Скажем, что две вершины неориентированного графа эквивалентны [3], если в графе можно провести путь, соединяющий эти вершины. Построим алгоритм разбиения множества вершин графа на классы эквивалентности, называемые компонентами связности.

На шаге 1 рассмотрим первую вершину (обозначена 1) и составим компоненту связности $K_1 = \{1\}$. На шаге $t-1$ вершины $1, \dots, t-1$ разбиты на компоненты связности $K_i,\ i \in J$:

$$K_i \cap K_j = \varnothing,\ i \neq j,\ i,j \in J,\ \bigcup_{i \in J} K_i = \{1, \dots, t-1\}.$$

На шаге t рассматриваем вершину t, вычисляем $c_i = \max\limits_{j \in K_i} d_{jt},\ i \in J$, и полагаем

$$K_i := K_i,\ i \in J/I,\ K_t := \{t\} \cup \left[\bigcup_{i \in I} K_i\right],\ J := (J/I) \cup \{t\},$$

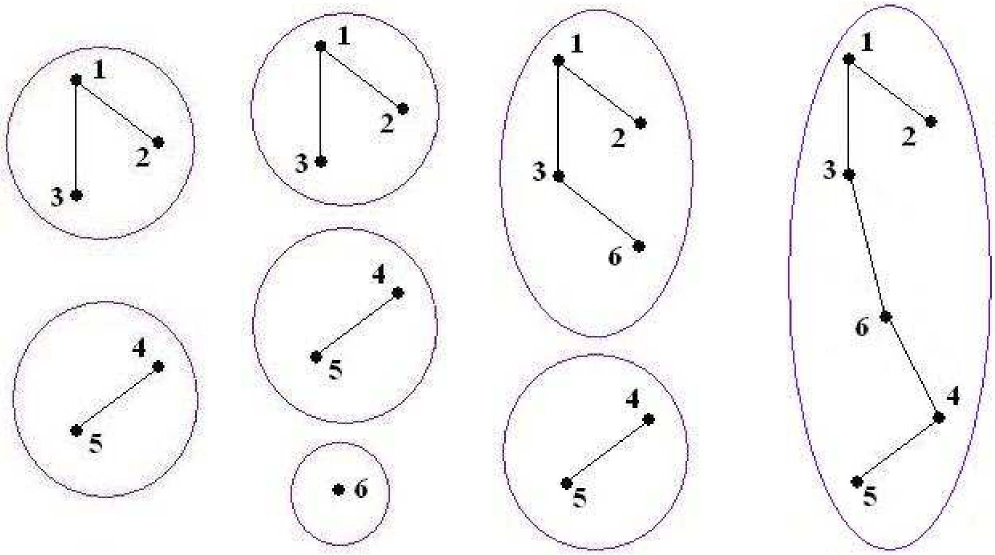
где $I = \{i \in J :\ c_i = 1\}$.

Ниже приведена графическая визуализация одного шага работы алгоритма и описаны компоненты связности в рассматриваемых случаях. Пусть на 5 шаге (рис. 1) были определены 2 компоненты связности $K_3 = \{1, 2, 3\},\ K_5 = \{4, 5\}$. На 6 шаге приведены возможные варианты факторизации: 1 вариант - вершина 6 не соединена ребром ни с одной из вершин $\{1, 2, 3, 4, 5\}$; 2 вариант - вершина 6 соединена ребром с вершиной 3; 3 вариант - вершина 6 соединена ребром с вершинами $3, 4$.

2. ФАКТОРИЗАЦИЯ ВЕРШИН ОРИЕНТИРОВАННОГО ГРАФА

Рассмотрим ориентированный граф с конечным множеством вершин $V = \{1, \dots, n\}$ и ребер. Зададим граф матрицей смежности $||d_{ij}||_{i,j=1}^n$, при этом положим $d_{ii} = 1,\ 1 \leqslant i \leqslant n$.

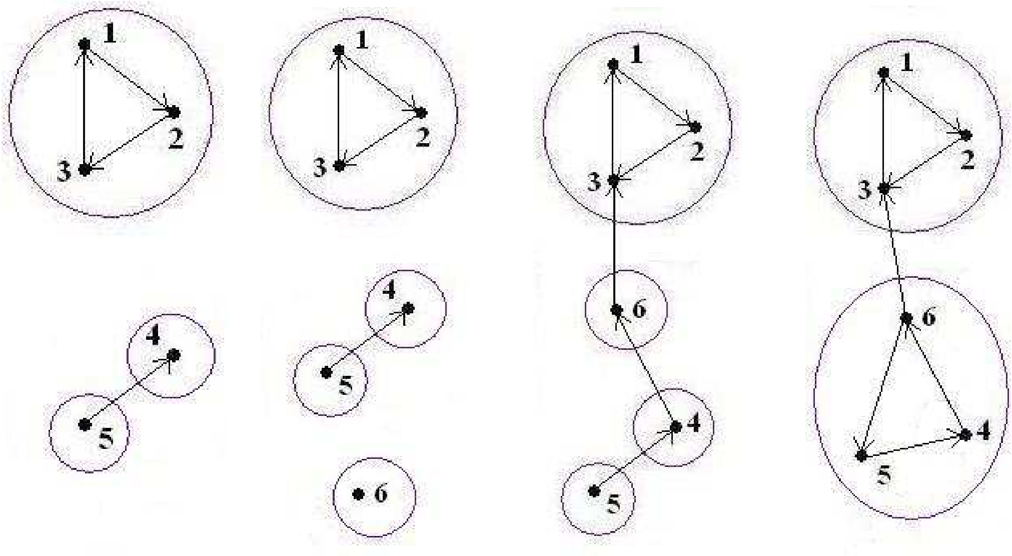
5-ый шаг	1 вариант 6-го шага	2 вариант 6-го шага	3 вариант 6-го шага



$K_3 = \{1,2,3\}$ $K_5 = \{4,5\}$	$K_3 = \{1,2,3\}$ $K_5 = \{4,5\}$ $K_6 = \{6\}$	$K_6 = \{1,2,3,6\}$ $K_5 = \{4,5\}$	$K_6 = \{1,2,3,4,5,6\}$

Рис. 1. *5 и 6 шаг работы алгоритма.*

5-ый шаг	1 вариант 6-го шага	2 вариант 6-го шага	3 вариант 6-го шага



$[3] = \{1,2,3\}$ $[4] = \{4\}$ $[5] = \{5\}$ $[5] \succeq [4]$	$[3] = \{1,2,3\}$ $[4] = \{4\}$ $[5] = \{5\}$ $[6] = \{6\}$ $[5] \succeq [4]$	$[3] = \{1,2,3\}$ $[4] = \{4\}$ $[5] = \{5\}$ $[6] = \{6\}$ $[5] \succeq [4] \succeq [6] \succeq [3]$	$[3] = \{1,2,3\}$ $[6] = \{4,5,6\}$ $[6] \succeq [3]$

Рис. 2. *5 и 6 шаг работы алгоритма.*

Скажем, что две вершины ориентированного графа эквивалентны [3], если существует цикл, содержащий эти вершины. Разобьем множество V на классы эквивалентности (компоненты сильной связности) и обозначим $[V]$ множество этих классов. На множестве $[V]$ определим бинарное отношение : $[v_1] \succeq [v_2],\ [v_1],\ [v_2] \in [V]$, если существует путь из вершины класса $[v_1]$ в вершину класса $[v_2]$. Очевидно, что это бинарное отношение рефлексивно, транзитивно и антисимметрично и потому является отношением частичного порядка [4, § 4]. Построим алгоритм разбиения множества вершин графа на классы эквивалентности с определением отношения частичного порядка между ними.

На шаге 1 имеется единственная вершина 1, образующая кластер [1], и множество кластеров $K = \{[1]\}$. Введем матрицу $a = ||a([p],[q])||_{[p],[q]\in K}$, характеризующую отношение частичного порядка " $\succeq$ " : $a([p],[q]) = 1$, если $[p] \succeq [q]$, иначе $a([p],[q]) = 0$. На шаге 1 эта матрица определяется равенством $a([1],[1]) = 1$.

Пусть на шаге $t-1$ множество кластеров K образуют разбиение множества вершин $\{1,\dots,t-1\}$ на не пересекающиеся подмножества и задана матрица a. Каждый кластер $[k] \in K$ индексируется максимальным номером вершины k, входящей в него. На шаге t для вершины t определим множества

$$P = \{k \in \{1,\dots,t-1\} :\ d_{tk} = 1\},\ Q = \{k \in \{1,\dots,t-1\} :\ d_{kt} = 1\}.$$

Каждой вершине множеств $P,\ Q$ сопоставляется кластер. Обозначим $[P],\ [Q]$ множества кластеров, соответствующих множествам вершин из P, Q, соответственно, и определим

$$K_P = \bigcup_{[p]\in[P]} \{[k] \in K :\ a([p],[k]) = 1\},\ K_Q = \bigcup_{[q]\in[Q]} \{[k] \in K :\ a([k],[q]) = 1\},$$

$$A = K_P \cap K_Q,\ A_1 = K_P \setminus A,\ A_2 = K_Q \setminus A,\ B = K \setminus (A \cup A_1 \cup A_2)\,.$$

Новая вершина t и вершины множества A образуют новый кластер, а также преобразуется множество кластеров K

$$[t] := \{t\} \cup A,\ K := (K \setminus A) \cup \{[t]\}.$$

Пересчет матрицы a на обновленном множестве кластеров K производится по правилу:

$$a([t],[i]) := 1,\ [i] \in A_1 \cup \{[t]\},\ a([i],[j]) := 1,\ [i] \in A_2,\ [j] \in A_1 \cup \{[t]\},$$

$$a([i],[j]) := 0,\ [i] \in A_1,\ [j] \in A_2 \cup \{[t]\} \cup B,\ a([i],[j]) := 0,\ [j] \in A_2,\ [i] \in B \cup \{[t]\},$$

$$a([t],[i]) = a([i],[t]) := 0,\ [i] \in B.$$

Все остальные значения матрицы a совпадают с определёнными на предыдущем шаге $t-1$. Матрица a имеет следующую клетчатую структуру:

<table>
<tr><td>матрица a</td><td>кластеры мн-ва A_1</td><td>кластеры мн-ва $[t]$</td><td>кластеры мн-ва A_2</td><td>кластеры мн-ва B</td></tr>
<tr><td>кластеры мн-ва A_1</td><td>значения на шаге $t-1$</td><td>0</td><td colspan="2" rowspan="2">0</td></tr>
<tr><td>кластеры мн-ва $[t]$</td><td colspan="2" rowspan="2">1</td></tr>
<tr><td>кластеры мн-ва A_2</td><td colspan="2">значения на шаге $t-1$</td></tr>
<tr><td>кластеры мн-ва B</td><td>значения на шаге $t-1$</td><td colspan="2">0</td><td>значения на шаге $t-1$</td></tr>
</table>

На рисунке 2 приведена графическая визуализация одного шага работы алгоритма и описаны кластеры в рассматриваемых случаях. Пусть на 5 шаге (рис. 2) множество вершин графа было разбито на 3 кластера $[3] = \{1, 2, 3\}$, $[4] = \{4\}$, $[5] = \{4\}$. На 6 шаге приведены возможные варианты факторизации: 1 вариант - вершина 6 не соединена ребром ни с одной из вершин $\{1, 2, 3, 4, 5\}$; 2 вариант - вершина 6 соединена ребром с вершиной $3, 4$; 3 вариант - вершина 6 соединена ребром с вершинами $3, 4, 5$.

СПИСОК ЛИТЕРАТУРЫ

1. Алгоритмы. Построение и анализ / Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн. — М.: Лаборатория базовых знаний, 2004. — 955 с.

2. Седжвик, Р. Фундаментальные алгоритмы на C++. Алгоритмы на графах: Пер с англ. / Р. Седжвик. — СПб: ООО «ДиаСофтЮП», 2002. — 496 с.

3. Оре, О. Теория графов / О. Оре. — М.: Наука, 1980. — 336 с.

4. Курош, А. Г. Лекции по общей алгебре / А. Г. Курош. — М.: Физматлит, 1962. — 431 с.

REFERENCES

1. Kormen T., Leizerson Ch., Rivest R., Shtain K. Algorithms. Construction and analysis. [Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы. Построение и анализ]. Moscow: Laboratory of basic knowledges, 2004, 955 p.

2. Sedgvik R. Fundamental algorithms on C++. Algorithms on graphs. [Седжвик Р. Фундаментальные алгоритмы на C++. Алгоритмы на графах]. Saint-Petersberg, 2002, 496 p.

3. Ore O. Graphs theory. [Оре О. Теория графов]. Moscow: Nauka, 1980, 336 p.

4. Kurosh A. G. Lectures on general algebra. [Курош А. Г. Лекции по общей алгебре]. Moscow: Fizmatlit, 1962, 431 p.

Цициашвили Г.Ш., профессор ДВФУ, зав. лабораторией ИПМ ДВО РАН, о. Русский, Кампус ДВФУ, Россия
E-mail: guram@iam.dvo.ru

Tsitsiashvili G.Sh., professor of FEFU, head of IAM FEB RAS laboratory, island Russian, Campus FEFU, Russia
E-mail: guram@iam.dvo.ru

Осипова М.А., доцент ДВФУ, н.с. ИПМ ДВО РАН, о. Русский, Кампус ДВФУ, Россия
E-mail: mao1975@list.ru

Osipova M.A., associate professor of FEFU, research of IAM FEB RAS, island Russian, Campus FEFU, Russia
E-mail: mao1975@list.ru

Лосев А.С., доцент ДВФУ, н.с. ИПМ ДВО РАН, о. Русский, Кампус ДВФУ, Россия
E-mail: alexax@bk.ru

Losev A.S., associate professor of FEFU, research of IAM FEB RAS, island Russian, Campus FEFU, Russia
E-mail: alexax@bk.ru