Об алгебраических операциях на графах, сохраняющих степенную последовательность

М. И. Лашева

1. Введение

В настоящей работе получены оригинальные алгоритмы для решения известной задачи о реализациях графической последовательности, допускающие обобщения на случаи ориентированных графов и гиперграфов и использующие ограниченный ресурс памяти.

В известной литературе [3] эта задача формулируется следующим образом. Графическая последовательность — это последовательность степеней вершин некоторого графа, при этом ее реализация — любой граф, имеющий такую степенную последовательность. В общем случае графическая последовательность имеет несколько попарно неизоморфных реализаций. В установлении связи между графическими реализациями играет роль понятие переключения. Пусть $G = (V, E)$ — граф, a, b, c, d — четыре различные его вершины такие, что $\{a, b\}, \{c, d\} \in E$, $\{a, c\}, \{b, d\} \notin E$. Тогда говорят, что граф допускает переключение $\{\{a, b\}, \{c, d\}\}$, а результатом применения операции переключения $\{\{a, b\}, \{c, d\}\}$ является граф $G' = (V, E'), E' = (E \cup \{\{a, c\}, \{b, d\}\}) \setminus \{\{a, b\}, \{c, d\}\}$. Известен алгоритм, используя который при помощи операции переключения ребер из любого графа с заданной степенной последовательностью может быть получен любой другой граф с той же степенной последовательностью. Время работы указанного алгоритма растет по порядку как $n^2 \log_2 n$. При этом алгоритм существенно использует возможность нелинейного (по размеру задачи) расширения либо оперативной, либо внешней памяти.

В настоящей работе построен алгоритм, решающий для всех графов ту же задачу с фиксированной оперативной памятью, внешняя память которого растет линейно относительно размера задачи. Получены верхние оценки времени работы рассматриваемого алгоритма.

Операция переключения для графов используется при решении некоторых практических проблем. Например, при оптимизации свойств компьютерных сетей с заданным множеством провайдеров и ограничениями на коммутационные возможности каждого из них. Решение этой практической задачи с использованием предлагаемых алгоритмов не потребует изучения глобальных характеристик всей сети, а лишь знания ее локальных свойств.

2. Ограниченный по памяти алгоритм преобразования графов с заданной степенной последовательностью

2.1. Основные понятия

Определение 1 ([2]). Графом $G = (V, E)$ называется объект, состоящий из двух конечных множеств: V — называемого множеством вершин, и множества E неупорядоченных пар различных элементов из V, то есть $E \subseteq V \times V = \{\, \{u, v\} : u \in V, v \in V, u \neq v\,\}$, называемого множеством ребер.

Определение 2 ([2]). Граф $G^{'} = (V^{'}, E^{'})$ называется подграфом $G = (V, E)$, если $V^{'} \subseteq V, E^{'} \subseteq E$.

Определение 3 ([3]). Мультиграфом $G^{*} = (V^{*}, E^{*})$ называется объект, состоящий из конечного множества вершин V^{*} и семейства E^{*} подмножеств множества $V \times V$ ребер (то есть элементы множества $V \times V$ в E^{*} могут повторяться).

Определение 4 ([2]). Граф $G^{*'} = (V^{*'}, E^{*'})$ называется подграфом мультиграфа $G^{*} = (V^{*}, E^{*})$, если $V^{*'} \subseteq V^{*}, E^{*'} \subseteq E^{*}$.

Определение 5 ([2]). Два ребра мультиграфа называются кратными, если они инцидентны одной и той же паре вершин.

Определение 6 ([2]). Степенью вершины v некоторого графа (мультиграфа) называется число ребер, инцидентных ей.

Количество вершин во всем графе $G = (V, E)$ обозначим $|V|$.

Определение 7 ([3]). Степенной последовательностью вершин графа $G = (V, E)$, где $|V| = n$, назовем набор степеней всех вершин графа $G = (V, E)$, упорядоченный по невозрастанию. Степенную последовательность обозначим $d_1(n)$.

Пусть даны два графа $G_1 = (V_1, E_1)$ и $G_2 = (V_2, E_2)$, причем $|V_1| = |V_2|$, $|E_1| = |E_2|$.. Пусть существует взаимно однозначное отображение $f : V_1 \leftrightarrow V_2$ такое, что для любых двух вершин $v_1 \in V_1, v_2 \in V_1$ выполняется условие: $\{v_1, v_2\} \in E_1 \Leftrightarrow \{f(v_1), f(v_2)\} \in E_2$. Такое отображение называется изоморфизмом графов [2] $G_1 = (V_1, E_1)$ и $G_2 = (V_2, E_2)$. Если для графов G_1 и G_2 существует изоморфизм f, то G_1 и G_2 называются изоморфными (обозначение $G_1 \cong G_2$).

Рассмотрим некоторое подмножество $A_1(d_1(n))$ множества графов $M_1(d_1(n))$ с занумерованными вершинами, обладающих заданной степенной последовательностью вершин $d_1(n)$. Определим операцию

$$\begin{aligned}\omega(A_1(d_1(n))) = \{G^{'} \mid \exists\, G = (V, E) \in A_1(d_1(n)),\\ \exists\, v_1, v_2, v_3, v_4 \in V,\ \{v_1, v_2\}, \{v_3, v_4\} \in E,\ \{v_1, v_3\}, \{v_2, v_4\} \notin E,\\ G^{'} = (V, (E \cup \{\, \{v_1, v_3\}, \{v_2, v_4\}\,\}) \setminus \{\, \{v_1, v_2\}, \{v_3, v_4\}\,\})\}.\end{aligned}$$

Очевидно, что операцию имеет смысл определять при $n \geqslant 4$. Кроме этого, отметим, что $\omega(A_1(d_1(n)))$ является подмножеством $M_1(d_1(n))$, то есть операция ω не меняет степенную последовательность вершин.

Будем говорить, что операция ω применима в графе G к некоторой паре ребер $\{v_{i_1}, v_{i_2}\}, \{v_{j_1}, v_{j_2}\}$, если в графе G не существует ребер $\{v_{i_1}, v_{j_1}\}, \{v_{i_2}, v_{j_2}\}$.

2.2. Постановка задачи. Описание алгоритма

Далее будем рассматривать множества графов с количеством вершин, не меньшим 4.

Обозначим Γ^2 множество пар $\{G_1, G_2\}$ неориентированных графов с занумерованными вершинами, у которых совпадают степенные последовательности вершин.

Рассмотрим пару графов $\{G_1, G_2\} \in \Gamma^2$,

$$G_1 = (V_1, E_1), G_2 = (V_2, E_2), |V_1| = |V_2| = n.$$

Известна следующая теорема [3].

Теорема 1. *Существуют целые неотрицательные числа N_1 и N_2, и найдутся графы G_1' и G_2' такие, что*

$$G_i' \in \omega^{N_i}(\{G_i\}) = \underbrace{\omega(\omega(\ldots\omega(\{G_i\})))}_{N_i},\ i = 1, 2,$$

и

$$G_1' \cong G_2'$$

(считаем по определению $\omega^0(M) = M$ для некоторого множества M).

Так как графы $G_1 = (V_1, E_1)$ и $G_2 = (V_2, E_2)$ имеют одинаковую степенную последовательность вершин, то биекция $r : V_2 \to V_1$, сопоставляющая каждой вершине v_1 графа $G_1 = (V_1, E_1)$ вершину v_2 графа $G_2 = (V_2, E_2)$ с тем же номером, сохраняет степень вершины.

Определим мультиграф G_{12} следующим образом: $G_{12} = (V_{12}, E_{12})$, где $V_{12} = V_1$, E_{12} — мультимножество, являющееся объединением мультимножеств E_1, E_2', где

$$E_2' = \left\{ \{r(v^{'}), r(v^{''})\} | \{v^{'}, v^{''}\} \in E_2 \right\}.$$

Определим в мультиграфе G_{12} подграфы $G^1 = (V_{12}, E_1)$, $G^2 = (V_{12}, E_2')$. Заметим, что подграф G^1 и граф G_1 (подграф G^2 и граф G_2) изоморфны. Таким образом, степень любой вершины мультиграфа G_{12} является удвоенной степенью вершины с тем же номером графа G_1. Обозначим степенную последовательность мультиграфа G_{12} — $d_2(n)$.

Выделим следующие свойства графа G_{12}.

Свойство 1. Для любой вершины мультиграфа G_{12} количество ребер, инцидентных ей и принадлежащих множеству E_1, равно количеству ребер, инцидентных ей и принадлежащих множеству $E_2^{'}$.

Свойство 2. Мультиграф G_{12} может иметь кратные ребра (кратности не выше 2), причем если e_1, e_2 — кратные ребра G_{12}, то $e_1 \in E_1$, $e_2 \in E_2'$ или $e_1 \in E_2'$, $e_2 \in E_1$.

Рассмотрим некоторое подмножество $A_2(d_2(n))$ множества мультиграфов $M_2(d_2(n))$ с занумерованными вершинами, обладающих степенной последовательностью $d_2(n)$ и удовлетворяющих свойствам 1 и 2. Определим операцию

$$\omega(A_2(d_2(n))) = \{G'_{12} \mid \exists\, G_{12} = (V_{12}, E_{12}) \in M_2(d_2(n)),$$
$$\exists\, v_1, v_2, v_3, v_4 \in V_{12},\ \{v_1, v_2\}, \{v_3, v_4\} \in E_1,\ \{v_1, v_3\}, \{v_2, v_4\} \notin E_1$$

(или $\{v_1, v_2\}, \{v_3, v_4\} \in E'_2,\ \{v_1, v_3\}, \{v_2, v_4\} \notin E'_2$),

$$G'_{12} = (V_{12}, (E_1 \cup \{\, \{v_1, v_3\}, \{v_2, v_4\}\, \}) \setminus \{\, \{v_1, v_2\}, \{v_3, v_4\}\, \})$$

(или $G'_{12} = (V_{12}, (E'_2 \cup \{\, \{v_1, v_3\}, \{v_2, v_4\}\, \}) \setminus \{\, \{v_1, v_2\}, \{v_3, v_4\}\, \}))\}$.

Будем говорить, что операция ω эффективно применима в мультиграфе G_{12} к некоторой паре ребер $\{v_{i_1}, v_{i_2}\} \in E_1, \{v_{j_1}, v_{j_2}\} \in E_1$ (или $\{v_{i_1}, v_{i_2}\} \in E'_2, \{v_{j_1}, v_{j_2}\} \in E'_2$), если в мультиграфе G_{12} не существует ребер $\{v_{i_1}, v_{j_1}\} \in E_1, \{v_{i_2}, v_{j_2}\} \in E_1$ ($\{v_{i_1}, v_{i_2}\} \in E'_2, \{v_{j_1}, v_{j_2}\} \in E'_2$), и при удалении ребер $\{v_{i_1}, v_{i_2}\} \in E_1, \{v_{j_1}, v_{j_2}\} \in E_1$ ($\{v_{i_1}, v_{i_2}\} \in E'_2, \{v_{j_1}, v_{j_2}\} \in E'_2$) и добавлении ребер $\{v_{i_1}, v_{j_1}\} \in E_1, \{v_{i_2}, v_{j_2}\} \in E_1$ ($\{v_{i_1}, v_{i_2}\} \in E'_2, \{v_{j_1}, v_{j_2}\} \in E'_2$) в полученном мультиграфе G'_{12} количество кратных ребер увеличивается.

Перед описанием алгоритма докажем следующие леммы.

Лемма 1. *Если существуют не кратные ребра*

$$\{v_{j_1}, v_{j_2}\} \in E'_2, \{v_{j_2}, v_{j_3}\} \in E_1,$$
$$\{v_{j_1}, v_{j_3}\} \in E_1, j_1 \neq j_2, j_2 \neq j_3,$$

то в мультиграфе $G_{12} = (V_{12}, E_{12})$ *эффективно применима операция* ω.

Доказательство.

Шаг 1. Существует не кратное ребро $\{v_{j_3}, v_{j_4}\} \in E'_2$ (следовательно, $j_4 \neq j_1, j_4 \neq j_2$).

Если не существует ребра $\{v_{j_1}, v_{j_4}\} \in E'_2$, то к ребрам $\{v_{j_1}, v_{j_2}\}$, $\{v_{j_3}, v_{j_4}\}$ эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_2}, v_{j_3}\}$. Остановка.

Если существуют кратные ребра $\{v_{j_1}, v_{j_4}\}$, то к ребрам $\{v_{j_1}, v_{j_4}\} \in E_1, \{v_{j_2}, v_{j_3}\}$ эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_1}, v_{j_2}\}$, и $\{v_{j_3}, v_{j_4}\}$. Остановка.

Если не существует ребра $\{v_{j_2}, v_{j_4}\} \in E_2^{'}$, то к ребрам $\{v_{j_1}, v_{j_2}\}$, $\{v_{j_3}, v_{j_4}\}$ эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_1}, v_{j_3}\}$. Остановка.

Если существует кратное ребро $\{v_{j_2}, v_{j_4}\}$, то к ребрам $\{v_{j_1}, v_{j_3}\}$, $\{v_{j_2}, v_{j_4}\} \in E_1$ эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_1}, v_{j_2}\}$, и $\{v_{j_3}, v_{j_4}\}$. Остановка.

Иначе переходим к шагу 2.

Шаг 2. Существует не кратное ребро $\{v_{j_4}, v_{j_5}\} \in E_1$ (следовательно, $j_5 \neq j_1, j_5 \neq j_2, j_5 \neq j_3$).

Если не существует ребра $\{v_{j_1}, v_{j_5}\} \in E_1$, то к ребрам $\{v_{j_1}, v_{j_3}\}, \{v_{j_4}, v_{j_5}\}$ эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_3}, v_{j_4}\}$. Остановка.

Если существует кратное ребро $\{v_{j_1}, v_{j_5}\}$, то к ребрам $\{v_{j_1}, v_{j_5}\} \in E_2', \{v_{j_3}, v_{j_4}\}$ эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_1}, v_{j_3}\}$, и $\{v_{j_4}, v_{j_5}\}$. Остановка.

Если не существует ребра $\{v_{j_2}, v_{j_5}\} \in E_1$, то к ребрам $\{v_{j_2}, v_{j_3}\}$, $\{v_{j_4}, v_{j_5}\}$ эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_3}, v_{j_4}\}$. Остановка.

Если существует кратное ребро $\{v_{j_2}, v_{j_5}\}$, то к ребрам $\{v_{j_2}, v_{j_5}\} \in E_2', \{v_{j_3}, v_{j_4}\}$ эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_2}, v_{j_3}\}$, и $\{v_{j_4}, v_{j_5}\}$. Остановка.

Если не существует ребра $\{v_{j_3}, v_{j_5}\} \in E_1$, то к ребрам $\{v_{j_2}, v_{j_3}\}$, $\{v_{j_4}, v_{j_5}\}$ (или к ребрам $\{v_{j_1}, v_{j_3}\}, \{v_{j_4}, v_{j_5}\}$) эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_2}, v_{j_4}\}$ (или $\{v_{j_1}, v_{j_4}\}$). Остановка.

Если существует кратное ребро $\{v_{j_3}, v_{j_5}\}$, то к ребрам $\{v_{j_2}, v_{j_4}\}$, $\{v_{j_3}, v_{j_5}\} \in E_2'$ (или к ребрам $\{v_{j_1}, v_{j_4}\}, \{v_{j_3}, v_{j_5}\} \in E_2'$) эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_2}, v_{j_3}\}$, и $\{v_{j_4}, v_{j_5}\}$ (или $\{v_{j_2}, v_{j_3}\}$, и $\{v_{j_4}, v_{j_5}\}$). Остановка.

Иначе переходим к шагу 3.

Шаг k. На предыдущем шаге получили подграф $G_{k+2} = (V_{k+2}, E_{k+2})$ графа G_{12} такой, что $V_{k+2} = \left\{ v_{j_1}, \ldots, v_{j_{k+2}} \right\}$, а множество E_{k+2} не кратных ребер удовлетворяет условию: $\{v_{j_i}, v_{j_n}\} \in E_1$, если n — нечетное, $\{v_{j_i}, v_{j_n}\} \in E_2^{'}$, если n — четное, где $n = \overline{1, k+2}$, $i = \overline{1, \ldots, n-1}$.

Существует не кратное ребро $\{v_{j_{k+2}}, v_{j_{k+3}}\} \in E_1$, если

$\{v_{j_{k+1}}, v_{j_{k+2}}\} \in E_2'$, и не кратное ребро $\{v_{j_{k+2}}, v_{j_{k+3}}\} \in E_2'$, если $\{v_{j_{k+1}}, v_{j_{k+2}}\} \in E_1$ (следовательно, $j_{k+3} \notin \{j_1, \dots, j_{k+1}\}$).

Для определенности предположим, что $\{v_{j_{k+2}}, v_{j_{k+3}}\} \in E_1$ (случай $\{v_{j_{k+2}}, v_{j_{k+3}}\} \in E_2'$ рассматривается аналогично).

Далее для каждого $i = 1, \dots, k$ проверяем следующие два условия.

Если не существует ребра $\{v_{j_i}, v_{j_{k+3}}\} \in E_1$, то к ребрам $\{v_{j_i}, v_{j_{k+1}}\}$, $\{v_{j_{k+2}}, v_{j_{k+3}}\}$ эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_{k+1}}, v_{j_{k+2}}\}$. Остановка.

Если существует кратное ребро $\{v_{j_i}, v_{j_{k+3}}\}$, то к ребрам $\{v_{j_i}, v_{j_{k+3}}\} \in E_2', \{v_{j_{k+1}}, v_{j_{k+2}}\}$ эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_i}, v_{j_{k+1}}\}$, и $\{v_{j_{k+2}}, v_{j_{k+3}}\}$. Остановка.

Если не существует ребра $\{v_{j_{k+1}}, v_{j_{k+3}}\} \in E_1$, то к ребрам $\{v_{j_k}, v_{j_{k+1}}\}$, $\{v_{j_{k+2}}, v_{j_{k+3}}\}$, эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_k}, v_{j_{k+2}}\}$. Остановка.

Если существует кратное ребро $\{v_{j_{k+1}}, v_{j_{k+3}}\}$, то к ребрам $\{v_{j_{k+1}}, v_{j_{k+3}}\} \in E_2', \{v_{j_k}, v_{j_{k+2}}\}$, эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_k}, v_{j_{k+1}}\}$, и $\{v_{j_{k+2}}, v_{j_{k+3}}\}$. Остановка.

Иначе $k := k + 1$ и переходим к шагу k.

Рассмотрим не кратные ребра $\{v_{j_1}, v_{j_2}\} \in E_2'$, $\{v_{j_2}, v_{j_3}\} \in E_1$, $\{v_{j_1}, v_{j_3}\} \in E_1$, применяем к ним последовательно шаги $1, 2$ и так далее. Допустим, остановка произойдет на шаге n. Это значит, в мультиграфе G_{12} были найдены два ребра, к которым эффективно применима операция ω, в результате чего образуется пара кратных ребер, одно из которых принадлежит множеству E_1, а другое — множеству E_2') (или две пары кратных ребер). Иначе алгоритм находит бесконечную последовательность попарно различных подграфов мультиграфа G_{12}, что невозможно в силу конечности мультиграфа G_{12}.

Лемма 1 доказана.

Аналогично доказывается Лемма 2.

Лемма 2. *Если существуют не кратные ребра*

$$\{v_{j_1}, v_{j_2}\} \in E_1, \{v_{j_2}, v_{j_3}\} \in E_2',$$
$$\{v_{j_1}, v_{j_3}\} \in E_2', j_1 \neq j_2, j_2 \neq j_3,$$

то в мультиграфе $G_{12} = (V_{12}, E_{12})$ *эффективно применима операция* ω.

Лемма 3. *Если существуют не кратные ребра*

$$\{v_{j_1}, v_{j_2}\} \in E_1, \{v_{j_2}, v_{j_3}\} \in E_2', \{v_{j_3}, v_{j_4}\} \in E_1,\ j_1 \neq j_4,$$

и не существует не кратных ребер

$$\{v_{j_1}, v_{j_3}\}, \{v_{j_2}, v_{j_4}\},$$

то в графе G_{12} *эффективно применима операция* ω.

Доказательство.

Шаг 1. Если существует ребро $\{v_{j_1}, v_{j_4}\} \in E_2'$, (возможно, кратное), то к ребрам

$$\{v_1, v_{j_4}\}, \{v_{j_2}, v_{j_3}\}$$

эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_1}, v_{j_2}\}, \{v_{j_3}, v_{j_4}\}$. Остановка.

Если не существует ребра $\{v_{j_1}, v_{j_4}\} \in E_1$, то к ребрам

$$\{v_1, v_{j_2}\}, \{v_{j_3}, v_{j_4}\}$$

эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_2}, v_{j_3}\}$.

Иначе переходим к шагу 2.

Шаг 2. Существует не кратное ребро $\{v_{j_4}, v_{j_5}\} \in E_2'$ ($j_5 \neq j_1, j_5 \neq j_2, j_5 \neq j_3$).

Если существует не кратное ребро $\{v_{j_3}, v_{j_5}\}$, то к тройке ребер $\{v_{j_3}, v_{j_4}\}, \{v_{j_4}, v_{j_5}\}, \{v_{j_3}, v_{j_5}\}$ применяем лемму 1 или 2 и находим не менее одной пары кратных ребер. Остановка.

Если существует ребро $\{v_{j_2}, v_{j_5}\} \in E_1$, то к ребрам $\{v_{j_2}, v_{j_5}\}$, $\{v_{j_3}, v_{j_4}\}$ эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_2}, v_{j_3}\}, \{v_{j_4}, v_{j_5}\}$. Остановка.

Если не существует ребра $\{v_{j_2}, v_{j_5}\} \in E_2'$, то к ребрам $\{v_{j_2}, v_{j_3}\}, \{v_{j_4}, v_{j_5}\}$ эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_3}, v_{j_4}\}$. Остановка.

Иначе переходим к шагу 3.

Заметим при этом, что количество ребер, принадлежащих E_1 (обозначим n_1), может изменяться от 0 до 2, а количество ребер, принадлежащих E_2' (обозначим n_2), — от 2 до 4, при этом при

$n_1 = 2 \quad n_2 \geqslant 3$. Из этих рассуждений и свойства 1 графа G_{12} следует возможность перехода к следующему шагу алгоритма.

Шаг k. На предыдущем шаге получили подграф $G_{k+3} = (V_{k+3}, E_{k+3})$ графа G_{12} такой, что $V_{k+3} = \left\{ v_{j_1}, \ldots, v_{j_{k+3}} \right\}$, а множество E_{k+3} обязательно включает в себя не кратные ребра вида (далее без ограничения общности положим k четным числом): $\{v_{j_{2l+1}}, v_{j_{2m+2}}\} \in E_1$, и $\{v_{j_{2l+2}}, v_{j_{2m+3}}\} \in E_2'$, где $m = \overline{1, \lceil k/2 \rceil}$, $l = \overline{1, m}$,

и, кроме того, возможно, содержит некоторые из нижеперечисленных кратных ребер: $\{v_{j_m}, v_{j_{m+2}}\}$, где $m = \overline{1, k+1}$ и ребер, которые могут быть как кратными, так и не кратными: $\{v_{j_{2p+1}}, v_{j_{2q+3}}\}$, где $q = \overline{1, \lceil k/2 \rceil}$, $p = \overline{0, q-1}$, и $\{v_{j_{2p}}, v_{j_{2q+2}}\}$, где $q = \overline{1, \lceil k/2 \rceil}$, $p = \overline{1, q-1}$.

При этом количество ребер выходящих из вершины $v_{j_{k+3}}$ и принадлежащих E_2', по крайней мере на 1 больше количества таких ребер, принадлежащих E_1. Отсюда вытекают следующие рассуждения.

Существует не кратное ребро $\{v_{j_{k+3}}, v_{j_{k+4}}\} \in E_1$, (следовательно, $j_{k+4} \notin \{j_1, \ldots, j_{k+2}\}$).

Если существует не кратное ребро $\{v_{j_{k+2}}, v_{j_{k+4}}\}$, то к тройке ребер $\{v_{j_{k+2}}, v_{j_{k+3}}\}, \{v_{j_{k+3}}, v_{j_{k+4}}\}, \{v_{j_{k+2}}, v_{j_{k+4}}\}$ применяем лемму 1 или 2 и находим не менее одной пары кратных ребер. Остановка.

Иначе для каждого $i = 2j+1$, $j = \overline{0, \lceil k/2 \rceil}$ проверяем следующие два условия.

Если существует ребро $\{v_{j_i}, v_{j_{k+4}}\} \in E_2'$, то к ребрам $\{v_{j_i}, v_{j_{k+4}}\}$, $\{v_{j_{k+2}}, v_{j_{k+3}}\}$ эффективно применима операция ω. Получаем две пары кратных ребер $\{v_{j_i}, v_{j_{k+2}}\}, \{v_{j_{k+3}}, v_{j_{k+4}}\}$. Остановка.

Если не существует ребра $\{v_{j_i}, v_{j_{k+4}}\} \in E_1$, то к ребрам $\{v_{j_i}, v_{j_{k+2}}\}$, $\{v_{j_{k+3}}, v_{j_{k+4}}\}$ эффективно применима операция ω. Получаем пару кратных ребер $\{v_{j_{k+2}}, v_{j_{k+3}}\}$. Остановка.

Иначе $k := k+1$ и переходим к шагу k.

Рассмотрим не кратные ребра $\{v_{j_1}, v_{j_2}\} \in E_1, \{v_{j_2}, v_{j_3}\} \in E_2'$, $\{v_{j_3}, v_{j_4}\} \in E_1$, применяем к ним последовательно шаги $1, 2$ и так далее. Допустим, остановка произойдет на шаге n. Это значит, в мультиграфе G_{12} были найдены два ребра, к которым эффективно применима операция ω, в результате чего образуется пара кратных ребер, одно из которых принадлежит множеству E_1, а другое — множеству E_2') (или две пары кратных ребер). Иначе алгоритм находит бесконечную последовательность попарно различных подграфов мультиграфа G_{12}, что невозможно в силу конечности мультиграфа G_{12}.

Лемма 3 доказана.

Аналогично доказывается Лемма 4.

Лемма 4. *Если существуют не кратные ребра*

$$\{v_{j_1}, v_{j_2}\} \in E_2', \{v_{j_2}, v_{j_3}\} \in E_1, \{v_{j_3}, v_{j_4}\} \in E_2',\ j_1 \neq j_4,$$

и не существует не кратных ребер

$$\{v_{j_1}, v_{j_3}\}, \{v_{j_2}, v_{j_4}\},$$

то в графе G_{12} эффективно применима операция ω.

Из Лемм 1–4 следует алгоритм нахождения пары ребер в мультиграфе G_{12}, к которым эффективно применима операция ω.

Более точно, верна следующая лемма.

Лемма 5. *Существует алгоритм, который для любого не кратного ребра мультиграфа G_{12} с заданной нумерацией вершин строит простую цепь с началом в одной из вершин этого ребра, определяющую пару ребер, к которым эффективно применима операция ω.*

Доказательство. Рассмотрим произвольное ребро $\{v_{j_1}, v_{j_2}\} \in E_1$ ($\{v_{j_1}, v_{j_2}\} \in E_2'$), не являющееся кратным. По свойству 1 существует не кратное ребро

$$\{v_{j_2}, v_{j_3}\} \in E_2' \ (\{v_{j_2}, v_{j_3}\} \in E_1),$$

а также существует не кратное ребро

$$\{v_{j_3}, v_{j_4}\} \in E_1 \ (\{v_{j_3}, v_{j_4}\} \in E_2').$$

1) Если $j_4 = j_1$, то используя лемму 1(2), найдем простую цепь

$$v_{j_1}, \{v_{j_1}, v_{j_2}\}, v_{j_2}, \{v_{j_2}, v_{j_3}\}, v_{j_3}, \{v_{j_3}, v_{j_4}\}, v_{j_4}, \ldots, v_{j_k},\ k \leqslant n$$

и пару ребер, к которым эффективно применима операция ω.

2) Если $j_4 \neq j_1$, то используя лемму 3(4), найдем простую цепь

$$v_{j_1}, \{v_{j_1}, v_{j_2}\}, v_{j_2}, \{v_{j_2}, v_{j_3}\}, v_{j_3}, \{v_{j_3}, v_{j_4}\}, v_{j_4}, \ldots, v_{j_l},\ l \leqslant n$$

и пару ребер, к которым эффективно применима операция ω.

Лемма 5 доказана.

Замечание. Удлинение простой цепи в результате добавления ребра является результатом анализа окрестности вершины мультиграфа, инцидентной данному ребру. Окрестность любой вершины мультиграфа, состоящего из n вершин, содержит не более $n-1$ ребер, а простая цепь в таком мультиграфе состоит не более, чем из $n-1$ ребер, следовательно, алгоритм Леммы 5 заканчивает свою работу не более, чем за n^2 шагов.

Теперь будем преобразовывать мультиграф $G_{12} = (V_{12}, E_{12})$, $|V_{12}| = n$, и соответствующие графы $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$ с заданной нумерацией вершин в соответствии с Общим Алгоритмом.

Общий Алгоритм.

$G'_{12} := G_{12}$

$G'_1 := G_1, G'_2 := G_2$

Цикл:

Шаг 1. Если существует пара кратных ребер $\{v_i, v_j\}$, то определим множество

$$E'_{12} := (E_1 \setminus \{v_i, v_j\}) \cup (E'_2 \setminus \{v_i, v_j\})$$

и перейдем к мультиграфу $G'_{12} := (V_{12}, E'_{12})$. Графы G'_1, G'_2 остаются без изменений.

Если мультиграф G'_{12} пуст, то алгоритм заканчивает работу.

Переход к шагу 1.

Если в мультиграфе G'_{12} не существует пары кратных ребер $\{v_i, v_j\}$, то переход к шагу 2.

Шаг 2. Так как мультиграф G'_{12} непуст, существует ребро

$$\{v_i, v_j\} \in E_1 \ (\{v_i, v_j\} \in E'_2),$$

не являющееся кратным. По Лемме 5 найдем пару ребер, к которым эффективно применима операция ω и применим ее в мультиграфе G'_{12} и к соответствующим ребрам в графе G'_1 (G'_2). Обозначим полученные мультиграф и пару графов, соответственно, G'_{12} и G'_1 (G'_2).

Переход к шагу 1.

Для Общего Алгоритма справедлива следующая теорема.

Теорема 2. *Общий Алгоритм по данной паре графов $\{G_1, G_2\} \in \Gamma^2$ строит изоморфные графы G'_1 и G'_2.*

Доказательство. При работе алгоритма после каждого шага 2 во вновь полученном мультиграфе возрастает количество кратных ребер, а общее число ребер не увеличивается. Поэтому после выполнения каждого цикла алгоритма (кроме, возможно, первого) число ребер мультиграфа G'_{12} уменьшается не менее, чем на 2. Следовательно, алгоритм работает конечное число шагов. В результате получим: $G'_{12} = (V_{12}, E'_{12}) \in \omega^N(\{G_{12}\})$ для некоторого N; все ребра мультиграфа G'_{12} — удалены; соответствующие графы

$$G'_1 \in \omega^{N_1}(\{G_1\})$$

и

$$G'_2 \in \omega^{N_2}(\{G_2\}).$$

Причем, при очередном $i+1$ применении операции ω к мультиграфу из множества $\omega^i(\{G_{12}\})$ происходит применение операции ω к соответствующему графу из множества $\omega^i(\{G_1\})$ или из множества $\omega^i(\{G_2\})$, то есть $N_1 + N_2 = N$. Так как в результате работы алгоритма множество ребер мультиграфа G'_{12} пусто, то для каждой пары вершин v_i, v_j ребро $\{v_i, v_j\}$ либо существует одновременно в графах G'_1, G'_2 и было удалено из мультиграфа G'_{12} на некотором шаге работы алгоритма, либо такого ребра не существует в графах G'_1, G'_2. Из этого следует, что G'_1 и G'_2 изоморфны.

Теорема 2 доказана.

Из этой теоремы получается известное [3] следствие.

Следствие 1. *Существуют целое неотрицательное число N и граф G'_1 такие, что графы G'_1,*

$$G_1^{'} \in \omega^N(\{G_1\}) = \underbrace{\omega(\omega(\ldots\omega(\{G_1\})))}_{N}$$

и G_2 изоморфны (при $N = 0 \quad \omega^N(\{G_1\}) = \{G_1\}$).

Далее рассмотрим некоторые характеристики изложенного алгоритма.

Под временной сложностью работы алгоритма будем подразумевать количество просмотренных за время его работы ребер.

Теорема 3. *Общий Алгоритм обладает следующими свойствами:*

1) *Временная сложность работы Общего Алгоритма на графе G с n вершинами, растет по порядку не быстрее, чем n^4.*
2) *Временная сложность работы Общего Алгоритма на графе G с n вершинами, степень каждой из которых не больше k, по порядку не превосходит k^2n^2.*
3) *Временная сложность работы Общего Алгоритма, обрабатывающего графы с n вершинами, степени которых не превосходят наперед заданной константы K, растет по порядку не быстрее, чем n^2.*

Доказательство. 1) После каждого выполнения цикла (кроме, возможно, первого) Общего Алгоритма число ребер мультиграфа G'_{12} уменьшается не менее, чем на 2. Следовательно, алгоритм заканчивает свою работу по порядку не более, чем за n^2 шагов. Принимая во внимание замечание к Лемме 5, заключаем, что общая временная сложность составляет по порядку не более, чем n^4 шагов.

2) После каждого выполнения цикла (кроме, возможно, первого) Общего Алгоритма число ребер мультиграфа G'_{12} уменьшается не менее, чем на 2. Следовательно, алгоритм заканчивает свою работу по порядку не более, чем за kn шагов. Из замечания к Лемме 5 следует, что для нахождении очередной пары кратных ребер требуется не более kn просмотров ребер. Из этого заключаем, что общая временная сложность составляет по порядку не более, чем k^2n^2 шагов.

3) Следует из п.2.

2.3. Понятие конечно-автоматного алгоритма

Рассмотрим вопрос о существовании конечного автомата A [1], который по паре $(G_1, G_2) \in \Gamma^2$ строил бы пару (G'_1, G'_2) такую, что $G'_i \in \omega^{N_i}(\{G_i\}), i = 1, 2$, и $G'_1 \cong G'_2$.

Определение 8. Конечно-автоматный алгоритм — это пара (A, T), где A — конечный автомат, T — некоторая двумерная таблица, каждый элемент t_{ij} которой является элементом некоторого конечного множества L. Автомат A применяется к таблице T в следующем смысле. В момент времени t автомат находится в некотором состоянии q_t. На вход автомата поступает некоторый элемент

$t_{ij} \in T$. Выходом является элемент из L. Он записывается в эту клетку; автомат перемещается в одну из клеток с координатами $(i-1,j),(i,j-1),(i+1,j),(i,j+1)$ (при условии, что такая клетка существует) или остается на месте в клетке с координатами (i,j) и переходит в некоторое состояние q_{t+1}. В начальном состоянии q_0 автомат находится в клетке с координатами $(1,1)$, и на вход автомату поступает элемент t_{11}. Работа автомата заканчивается, если он переходит в заключительное состояние $q^{'}$.

В этом пункте будет доказано, что Общий Алгоритм является конечно-автоматным.

Конечно-автоматный алгоритм работает по схеме: кодирование, обработка результатов кодирования некоторым конечным автоматом, декодирование.

Рассмотрим множество пятерок

$$L = \{\{c_1, c_2, I_1, I_2, I\},\ c_1 \in \{0,1,2,3,4,5\}, c_2 \in \{0,1,2,3,4,5\},$$
$$I_1 \in \{0,1\}, I_2 \in \{0,1\}, I \in \{0,1,2\}\}.$$

Блок кодирования конечно-автоматного алгоритма паре графов $(G_1,G_2) \in \Gamma^2, G_i = (V_i,E_i),\ |V_i| = n, i = 1,2$, взаимно однозначно сопоставляет таблицу специального вида — треугольник смежностей T размера n.

Определение 9. Треугольник смежностей размера n, сопоставленный паре графов $(G_1,G_2) \in \Gamma^2$, — это таблица с n строчками и n столбцами, $T = (t_{ij}), i = 1,2,\dots,n, j = i, i+1,\dots,n$, каждый элемент t_{ij} которой является элементом конечного множества L.

Треугольник смежностей T взаимно однозначно сопоставляется паре графов $(G_1,G_2) \in \Gamma^2$ следующим образом: для $i \leqslant j$ положим

$$t_{ij} = \{c_1, c_2, I_1, I_2, I\},$$

где

$c_1 = c_2 = 0$,

$I_k = 1$, если ребро $\{v_i, v_j\} \in E_k$; $I_k = 0$, — иначе, $k = 1,2$;

$I = 1$, если $i = j,\ i \neq 1, n,\ i \neq 1, n$, либо $i = 1, 1 < j < n$, либо $j = n, 1 < i < n$; $I = 2$, если $i = j = 1,\ i = j = n,\ i = 1, j = n$; $I = 0$ — иначе.

Автомат A перерабатывает треугольник смежностей T, в соответствии с приведенной ниже схемой, в треугольник смежностей T', по которому взаимно однозначно восстанавливается пара графов (G_1', G_2') такая, что $G_i' \in \omega^{N_i}(\{G_i\}), i = 1, 2$, и $G_1' \cong G_2'$.

Блок декодирования строит по T' такую пару (G_1', G_2') следующим образом: если

$$t_{ij} = \{c_1, c_2, I_1, I_2, I\},$$

то при $I_k = 1 \quad \{v_i, v_j\} \in E_k,\ k = 1, 2$, иначе — $\{v_i, v_j\} \notin E_k$.

Теорема 4. *Общий алгоритм, который по паре* $(G_1, G_2) \in \Gamma^2$ *строит пару* (G_1', G_2') *такую, что* $G_i' \in \omega^{N_i}(\{G_i\}), i = 1, 2$, *и* $G_1' \cong G_2'$, *является конечно-автоматным.*

Доказательство. Построим блок-схему Общего алгоритма (см. рис. 1).

Построим конечные автоматы, которые применяются к треугольнику смежностей T и реализуют следующие инструкции.

Инструкция 1. Начальное состояние 1. Находясь на некотором ребре $\{v_i, v_j\}$ мультиграфа, поменять цвет вершины v_i на c и вернуться на исходное ребро $\{v_i, v_j\}$.

Начальное состояние 1'. Находясь на некотором ребре $\{v_i, v_j\}$ мультиграфа, поменять цвет вершины v_j на c и вернуться на исходное ребро $\{v_i, v_j\}$ (соответствующая диаграмма Мура строится аналогично предыдущей с заменой R на U, U на R, L на D, D на L).

Инструкция 2. Находясь на некотором ребре $\{v_i, v_j\}$ мультиграфа, проверить некоторое условие $f : L \to \{0, 1\}$ для окрестности вершины v_i и вернуться на исходное ребро $\{v_i, v_j\}$, при этом выполнение условия происходит при $f = 1$.

Инструкция 3. Находясь на некотором ребре $\{v_i, v_j\}$ мультиграфа, осмотреть окрестность вершины v_i и перейти на ребро из этой окрестности, удовлетворяющее некоторому условию: условием является некоторая функция $f : L \to \{0, 1\}$, при этом выполнение условия происходит при $f = 1$.

Замечание. Для удобства конечные состояния на диаграмме Мура продублированы.

Инструкция 4. Обойти треугольник смежностей T в горизонтальном направлении, проверяя некоторое условие $f : L \to \{0, 1\}$ в каждой ячейке, при этом выполнение условия происходит при $f = 1$.

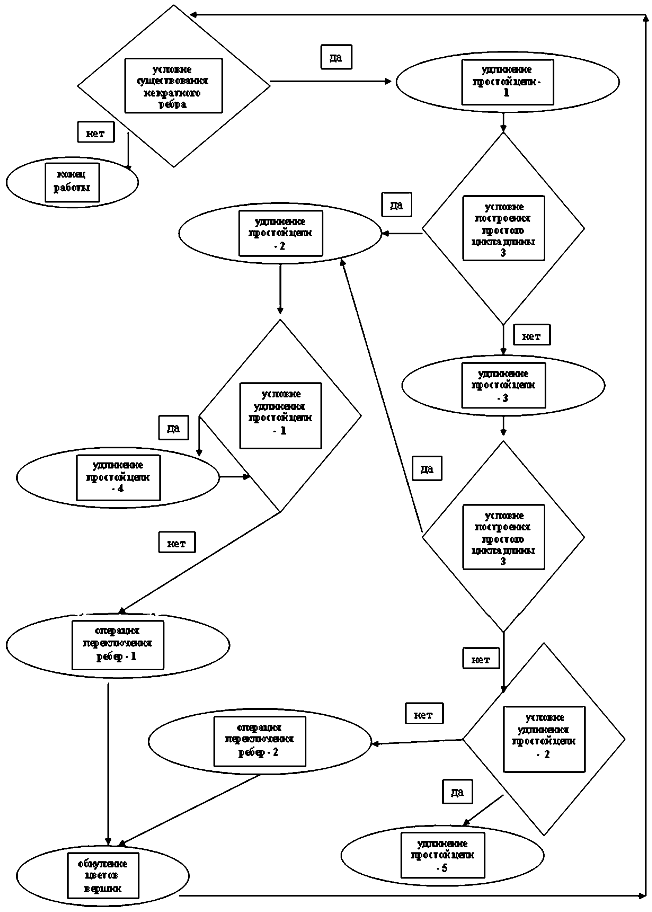


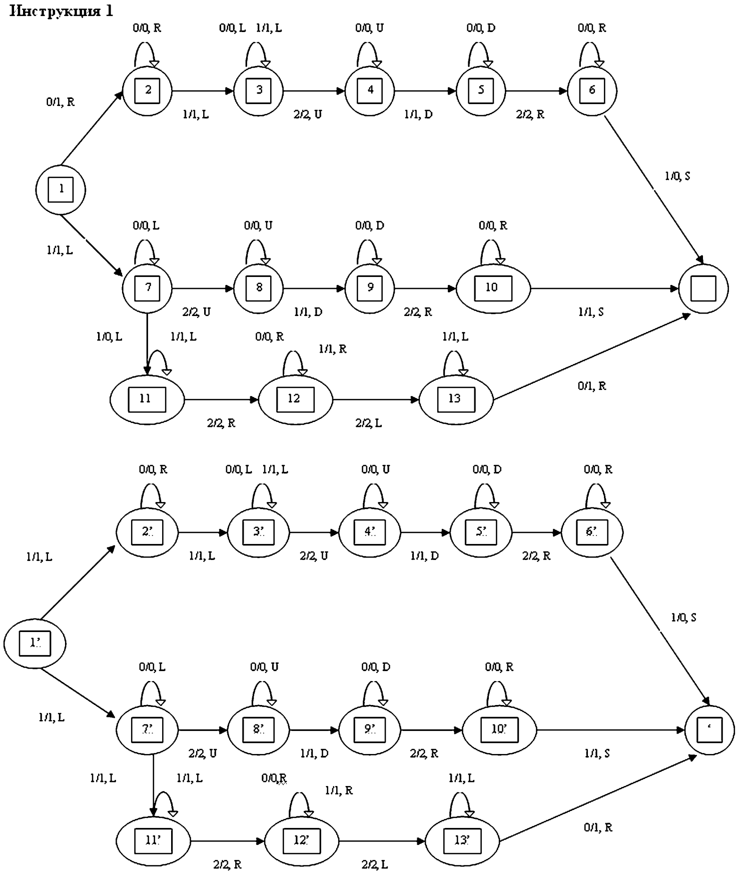
Рис. 1. Блок-схема Общего алгоритма.

Инструкция 5. Вернуться в ячейку с координатами (1, 1).

На диаграммах Мура соответствующих автоматов указаны стрелки с приписанными им строками двух видов.

1. Строка вида $n_1/n_2, M$, где $n_i \in \{0, 1, 2\}$, $i = 1, 2$, а $M \in \{R, L, U, D, S\}$, означает, что в случае поступления на вход автомата пятерки $t_{ij} = \{c_1, c_2, I_1, I_2, I\}$, где $I = n_1$, (в случае Инструкции 2 — при выполнении для этой пятерки заданного условия, если условие еще не было выполнено) автомат

Инструкция 1



1) записывает в эту ячейку пятерку $t'_{ij} = \{c'_1, c'_2, I_1, I_2, I'\}$, где $I' = n_2$,

$c'_1 = c,\ c'_2 = c_2$, если $M \in \{R, L\}$,

$c'_1 = c_1,\ c'_2 = c$, если $M \in \{U, D\}$,

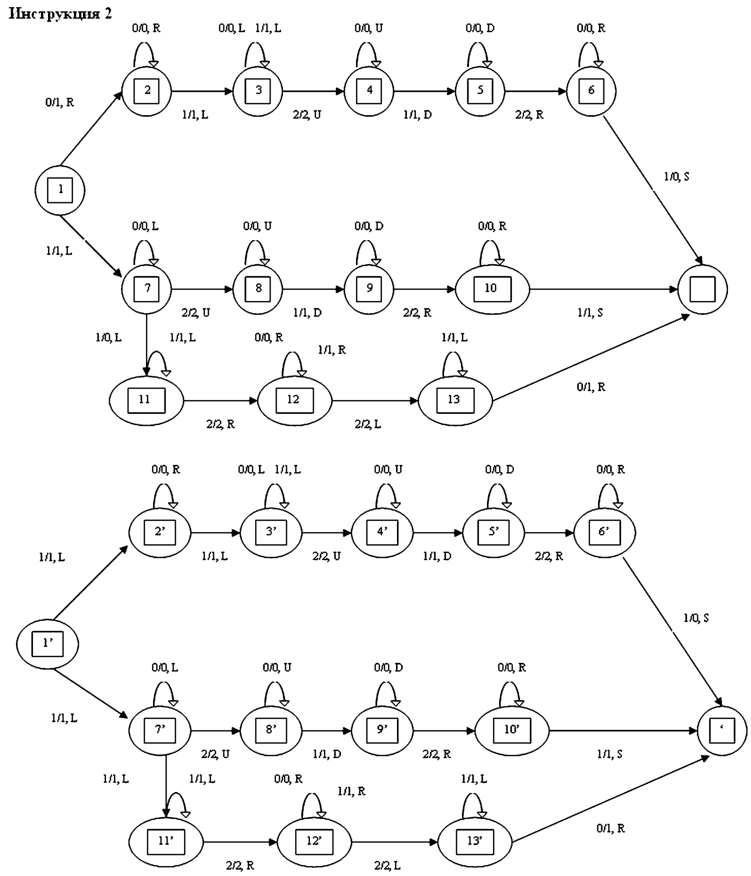
$c'_1 = c_1,\ c'_2 = c_2$, если $M = S$,

(в случае Инструкции 2 — автомат фиксирует значение 1 условия f и далее его не проверяет) и

2) перемещается:

вправо, если $M = R$,

влево, если $M = L$,



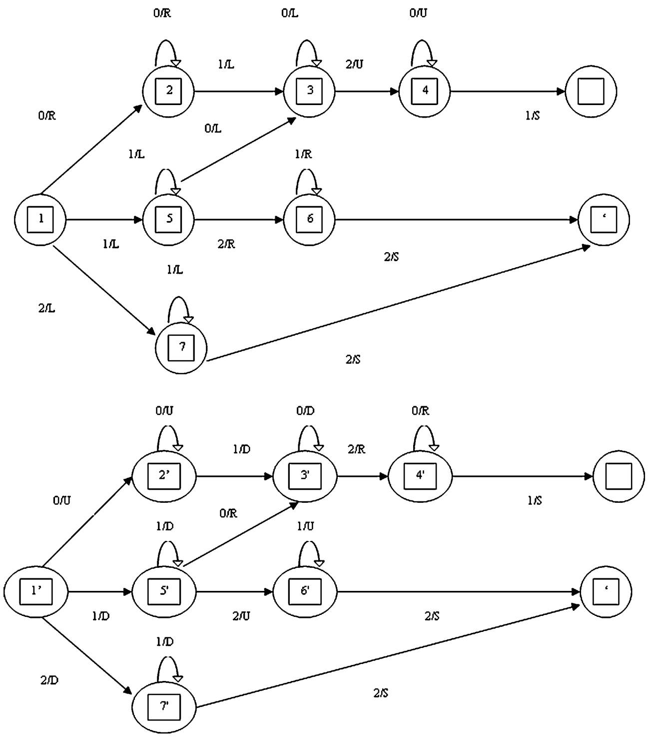
вверх, если $M = U$,

вниз, если $M = D$,

и остается на месте, если $M = S$;

2. Строка вида n/M, где $n \in \{0, 1, 2\}$, а $M \in \{R, L, U, D, S\}$, означает, что в случае поступления на вход автомата пятерки $t_{ij} = \{c_1, c_2, I_1, I_2, I\}$, где $I = n$, для которой не выполняется заданное условие, то есть $f(c_1, c_2, I_1, I_2, I) \neq 1$, автомат перемещается: вправо, если $M = R$, влево, если $M = L$, вверх, если $M = U$, вниз, если $M = D$, и остается на месте, если $M = S$, а также изменяет некоторым образом цвета c_1, c_2, если это дополнительно указано в описании алгоритма.

Инструкция 3

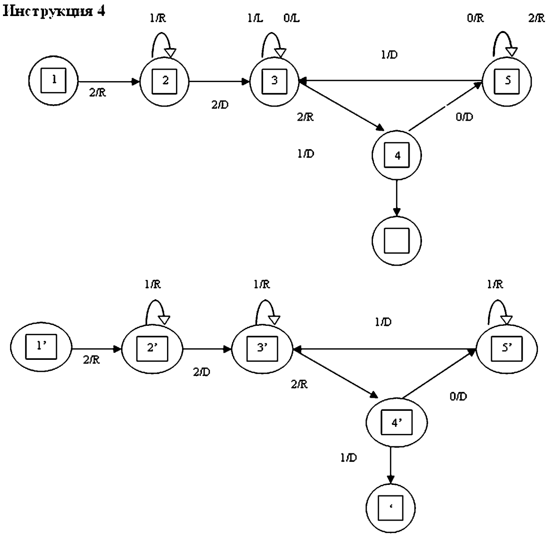


Описанием блоков блок-схемы Общего Алгоритма является последовательность Инструкций, при этом следует отметить, что диаграммы Мура Инструкций 1–3 состоят из двух аналогичных кусков, имеют два начальных и два конечных состояния. В описании последовательности Инструкций подразумевается, что если предыдущая работа автомата завершилась в простом конечном состоянии (конечном состоянии со штрихом), то следующая Инструкция выполняется в соответствии с диаграммой Мура с простым начальным состоянием (с начальным состоянием со штрихом), если явно не указано иное.

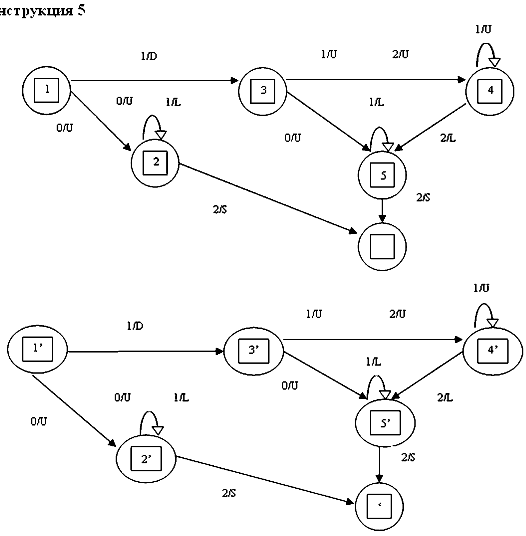
Теперь опишем работу каждого блока блок-схемы Общего Алгоритма с помощью Инструкций 1–4 и 1'–4'.

Общий Алгоритм начинает свою работу с блока проверки условия существования не кратного ребра.

Инструкция 4



Инструкция 5



Условие существования не кратного ребра.

Автомат работает по Инструкции 4, проверяя условие $f(c_1, c_2, I_1, I_2, I) = I_1 + I_2$. Обозначим поступившую при выполнении условия на вход автомата пятерку — $(c_1, c_2, I_1^*, I_2^*, I)$.

Удлинение простой цепи–1.
Автомат работает по Инструкции 1 (начальное состояние 1) с $c = 4$,
по Инструкции 1 (начальное состояние 1') с $c = 3$,
по Инструкции 3 с условием $f(c_1, c_2, I_1, I_2, I) = 1$ только при $c_1 = 3$, $c_2 = 0$, $I_1 \neq I_2$,
по Инструкции 1 (начальное состояние 1') с $c = 2$.

Условие построения простого цикла длины 3.
Автомат работает по Инструкции 2 с условием $f(c_1, c_2, I_1, I_2, I) = 1$ только при $c_1 = 2$, $c_2 = 4$, $I_1 \neq I_2$.

Удлинение простой цепи–2.
Автомат работает по Инструкции 3 с условием $f(c_1, c_2, I_1, I_2, I) = 1$ только при $c_1 = 2$, $c_2 = 4$, $I_1 \neq I_2$.
Автомат работает по Инструкции 1 (начальное состояние 1) с $c = 4$, затем по Инструкции 1 (начальное состояние 1') с $c = 2$ при $I_1 = I_1^*$, $I_2 = I_2^*$.
Автомат работает по Инструкции 5,
по Инструкции 4 с $f \equiv 0$, заменяя $c_i = 1, c_j = 5,\ i = 1, 2, j = 1, 2$ на 0,
по Инструкции 5,
по Инструкции 4 с $f = 1$ только при $c_1 = 4, c_2 = 2$ или $c_1 = 2, c_2 = 4$ и $I_1 \neq I_2$,
при $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$ по Инструкции 3 с условием $f(c_1, c_2, I_1, I_2, I) = 1$ только при $c_1 = 2,\ c_2 = 0,\ I_1 = I_1^*,\ I_2 = I_2^*,\ (I_1 \neq I_1^*,\ I_2 \neq I_2^*)$,
по Инструкции 1 (начальное состояние 1') с $c = 2$,
по Инструкции 1 (начальное состояние 1) с $c = 3$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 3,\ c_2 = 3$,
по Инструкции 1 (начальное состояние 1') с $c = 4$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 4,\ c_2 = 4$,
по Инструкции 1 (начальное состояние 1') с $c = 1$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 4,\ c_2 = 3$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 3,\ c_2 = 2$.

Условие удлинения простой цепи–1.
При $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$ автомат работает по Инструкции 2 с условием $f(c_1, c_2, I_1, I_2, I) = 1$ только при $c_1 = 2,\ c_2 = 1$, $I_1 = I_1^*,\ I_2 = I_2^*,\ (I_1 \neq I_1^*,\ I_2 \neq I_2^*)$ или при $c_1 = 2,\ c_2 = 4,\ I_1 = I_1^*$, $I_2 = I_2^*,\ (I_1 \neq I_1^*,\ I_2 \neq I_2^*)$

Удлинение простой цепи–3.
Автомат работает по Инструкции 3 с условием $f(c_1, c_2, I_1, I_2, I) = 1$ только при $c_1 = 2,\ c_2 = 0,\ I_1 = I_1^*,\ I_2 = I_2^*$,
по Инструкции 1 (начальное состояние 1') с $c = 2$,
по Инструкции 1 (начальное состояние 1) с $c = 3$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 3,\ c_2 = 3$,
по Инструкции 1 (начальное состояние 1') с $c = 4$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 4,\ c_2 = 4$,
по Инструкции 1 (начальное состояние 1') с $c = 1$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 4,\ c_2 = 3$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 3,\ c_2 = 2$.

Условие удлинения простой цепи–2.
При $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$ автомат работает по Инструкции 2 с условием $f(c_1, c_2, I_1, I_2, I) = 1$ только при $c_1 = 2,\ c_2 = 1$, $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$.

Удлинение простой цепи–4.
При $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$ автомат работает по Инструкции 3 с условием $f(c_1, c_2, I_1, I_2, I) = 1$ только при $c_1 = 2,\ c_2 = 0$, $I_1 = I_1^*,\ I_2 = I_2^*,\ (I_1 \neq I_1^*,\ I_2 \neq I_2^*)$,
по Инструкции 1 (начальное состояние 1') с $c = 2$,
по Инструкции 1 (начальное состояние 1) с $c = 3$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 3,\ c_2 = 3$,
по Инструкции 1 (начальное состояние 1') с $c = 4$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 4,\ c_2 = 4$,
по Инструкции 1 (начальное состояние 1') с $c = 1$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 4,\ c_2 = 3$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 3,\ c_2 = 2$.

Удлинение простой цепи–5.
Автомат работает по Инструкции 5,
по Инструкции 4 с $f \equiv 0$, заменяя $c_i = 5$ на $c_i = 1$ и $c_i = 1$ на $c_i = 5,\ i = 1, 2$,
При $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$ автомат работает по Инструкции 3 с условием $f(c_1, c_2, I_1, I_2, I) = 1$ только при $c_1 = 2,\ c_2 = 0$, $I_1 = I_1^*,\ I_2 = I_2^*,\ (I_1 \neq I_1^*,\ I_2 \neq I_2^*)$,
по Инструкции 1 (начальное состояние 1') с $c = 2$,
по Инструкции 1 (начальное состояние 1) с $c = 3$,

по Инструкции 3 с условием $f = 1$ только при $c_1 = 3,\ c_2 = 3$,
по Инструкции 1 (начальное состояние 1') с $c = 4$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 4,\ c_2 = 4$,
по Инструкции 1 (начальное состояние 1') с $c = 1$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 4,\ c_2 = 3$,
по Инструкции 3 с условием $f = 1$ только при $c_1 = 3,\ c_2 = 2$.

Операция переключения ребер–1.
При $c_1 = 2, c_2 = 1$ автомат работает следующим образом:
При $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$ автомат записывает в текущую ячейку пятерку $(c_1, c_2, I_1', I_2', I)$, такую что $I_1' \neq I_1^*,\ I_2' \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$
затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 1$, $c_2 = 4$,
автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 0, 0, I)$,
затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 4$, $c_2 = 3$,
автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 1, 1, I)$,
затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 3$, $c_2 = 2$,
автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 0, 0, I)$.
При $c_1 = 2, c_2 = 4$ автомат работает следующим образом:
При $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$ автомат записывает в текущую ячейку пятерку $(c_1, c_2, I_1', I_2', I)$, такую что $I_1' \neq I_1^*,\ I_2' \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$
затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 4$, $c_2 = 1$,
автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 0, 0, I)$,
затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 1$, $c_2 = 3$,
автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 1, 1, I)$,
затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 3$, $c_2 = 2$,

автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 0, 0, I)$.

Операция переключения ребер–2.

При $I_1 + I_2 = 0$ автомат работает следующим образом:

При $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$ автомат записывает в текущую ячейку пятерку $(c_1, c_2, I_1', I_2', I)$, такую что $I_1' \neq I_1^*,\ I_2' \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$

затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 1$, $c_2 = 4$,

автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 0, 0, I)$,

затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 4$, $c_2 = 3$,

автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 1, 1, I)$,

затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 3$, $c_2 = 2$,

автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 0, 0, I)$.

При $I_1 + I_2 = 1$ автомат работает следующим образом:

При $I_1 \neq I_1^*,\ I_2 \neq I_2^*,\ (I_1 = I_1^*,\ I_2 = I_2^*)$ автомат записывает в текущую ячейку пятерку $(c_1, c_2, 0, 0, I)$,

затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 1$, $c_2 = 4$,

автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 1, 1, I)$,

затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 1$, $c_2 = 3$,

автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 0, 0, I)$,

затем работает по Инструкции 3 с условием $f = 1$ только при $c_1 = 3$, $c_2 = 2$,

автомат записывает в найденную ячейку с пятеркой (c_1, c_2, I_1, I_2, I) пятерку $(c_1, c_2, 1, 1, I)$.

Обнуление цветов вершин.

Автомат работает по Инструкции 5,

затем по Инструкции 4 с условием $f = 1$, заменяя каждую поступившую на вход автомата пятерку (c_1, c_2, I_1, I_2, I) на пятерку $(0, 0, I_1, I_2, I)$.

Теорема доказана.

Общий Алгоритм является конечно-автоматным, поэтому размер задачи определим как количество ячеек треугольника смежностей, необходимого для задания входных данных Общего Алгоритма.

Теорема 5. *Объем памяти, необходимый для работы Общего Алгоритма, растет линейно относительно увеличения размера задачи.*

Доказательство. Пусть входными данными Общего алгоритма является пара $(G_1, G_2) \in \Gamma^2,\ G_i = (V_i, E_i), |V_i| = n, i = 1, 2$, тогда количество ячеек треугольника смежностей равно по порядку n^2 (следует из определения треугольника смежностей). Из доказательства предыдущей теоремы следует, что объем необходимой для работы Общего алгоритма памяти возрастает не более, чем на 1 ячейку по сравнению с объемом, необходимым для задания входных данных, а следовательно, растет линейно относительно увеличения размера задачи.

Теорема доказана.

3. Ограниченный по памяти алгоритм преобразования ориентированных графов с заданной степенной последовательностью

3.1. Основные понятия

Определение 10 ([2]). Ориентированным графом $G = (V, E)$ называется объект, состоящий из двух конечных множеств: V — называемого множеством вершин, и множества E упорядоченных пар различных элементов из V, то есть $E \subseteq V \times V = \{\, (u, v) : u \in V, v \in V, u \neq v \,\}$, называемого множеством дуг.

Определение 11 ([2]). Ориентированный граф $G' = (V', E')$ называется ориентированным подграфом $G = (V, E)$, если $V' \subseteq V, E' \subseteq E$.

Определение 12. Ориентированным мультиграфом $G^* = (V^*, E^*)$ называется объект, состоящий из конечного множества вершин V^*

и мультимножества дуг E^* (то есть множества, в которое каждый элемент входит с некоторой кратностью).

Определение 13 ([2]). Ориентированный граф $G^{*'} = (V^{*'}, E^{*'})$ называется подграфом ориентированного мультиграфа $G^* = (V^*, E^*)$, если $V^{*'} \subseteq V^*, E^{*'} \subseteq E^*$.

Определение 14 ([2]). Две дуги ориентированного мультиграфа называются кратными, если они являются входящими для одной и той же вершины и исходящими для одной и той же вершины.

Определение 15. Степенью вершины v некоторого ориентированного графа (мультиграфа) называется упорядоченная пара $\{d_{out}(v), d_{in}(v)\}$, где $d_{out}(v)$ — число дуг, исходящих из вершины v, а $d_{in}(v)$ — число дуг, входящих в вершину v.

Количество вершин во всем ориентированном графе $G = (V, E)$ обозначим $|V|$.

Определение 16. Степенной последовательностью вершин ориентированного графа $G = (V, E)$ назовем невозрастающую в лексикографическом смысле последовательность степеней всех вершин графа $G = (V, E)$. Обозначим ее $d_1(n)$.

Пусть даны два орграфа $G_1 = (V_1, E_1)$ и $G_2 = (V_2, E_2)$, причем $|V_1| = |V_2|$, $|E_1| = |E_2|$. Пусть существует взаимно однозначное отображение $f\colon V_1 \to V_2$ такое, что для любых двух вершин $v_1 \in V_1, v_2 \in V_1$ выполняется условие: $(v_1, v_2) \in E_1 \Leftrightarrow (f(v_1), f(v_2)) \in E_2$. Такое отображение называется изоморфизмом орграфов [2] $G_1 = (V_1, E_1)$ и $G_2 = (V_2, E_2)$. Если для орграфов G_1 и G_2 существует изоморфизм f, то G_1 и G_2 называются изоморфными (обозначение $G_1 \cong G_2$).

Рассмотрим некоторое подмножество $A_1(d_1(n))$ множества орграфов $M_1(d_1(n))$ с занумерованными вершинами, обладающих заданной степенной последовательностью вершин $d_1(n)$. Определим операцию

$$\begin{aligned}
\omega(A_1(d_1(n))) = \{G' |\, &\exists\, G = (V, E) \in A_1(d_1(n)),\\
&\exists\, v_1, v_2, v_3, v_4 \in V,\ (v_1, v_2), (v_3, v_4) \in E,\ (v_1, v_4), (v_3, v_2) \notin E,\\
&\quad G^{'} = (V, (E \cup \{\, (v_1, v_4), (v_3, v_2)\,\}) \setminus \{\, (v_1, v_2), (v_3, v_4)\,\})\}.
\end{aligned}$$

Очевидно, что операцию имеет смысл определять при $n \geqslant 4$. Кроме этого, отметим, что $\omega(A_1(d_1(n)))$ является подмножеством $M_1(d_1(n))$, то есть операция ω не меняет степенную последовательность вершин.

Будем говорить, что операция ω применима в орграфе G к некоторой паре дуг $(v_{i_1}, v_{i_2}), (v_{j_1}, v_{j_2})$, если в графе G не существует дуг $(v_{i_1}, v_{j_2}), (v_{j_1}, v_{i_2})$.

3.2. Постановка задачи. Описание алгоритма

Далее будем рассматривать множества графов с количеством вершин, не меньшим 4.

Обозначим Γ^2 множество пар $\{G_1, G_2\}$ орграфов с занумерованными вершинами, у которых совпадают степенные последовательности вершин.

Рассмотрим пару орграфов $\{G_1, G_2\} \in \Gamma^2$,

$$G_1 = (V_1, E_1), G_2 = (V_2, E_2), |V_1| = |V_2| = n.$$

Сформулируем следующую теорему (аналогичную теореме для неориентированных графов).

Теорема 6. *Существуют числа N_1 и N_2 и найдутся ориентированные графы G'_1 и G'_2 такие, что $G'_i \in \omega^{N_i}(\{G_i\}) = \underbrace{\omega(\omega(\ldots\omega(\{G_i\})))}_{N_i}$, $i = 1, 2$, и $G'_1 \cong G'_2$ (считаем по определению $\omega^0(M) = M$ для некоторого множества M).*

Так как $G_1 = (V_1, E_1)$ и $G_2 = (V_2, E_2)$ имеют одинаковую степенную последовательность вершин, то биекция $r : V_2 \to V_1$, сопоставляющая каждой вершине v_1 орграфа $G_1 = (V_1, E_1)$ вершину v_2 орграфа $G_2 = (V_2, E_2)$ с тем же номером, сохраняет степень вершины.

Определим ориентированный мультиграф G_{12} следующим образом: $G_{12} = (V_{12}, E_{12})$, где $V_{12} = V_1$, E_{12} — мультимножество, являющееся объединением мультимножеств E_1, E'_2, где $E'_2 = \Big\{ (r(v'), r(v'')) | (v', v'') \in E_2 \Big\}$. Определим в ориентированном мультиграфе G_{12} ориентированные подграфы $G^1 = (V_{12}, E_1)$, $G^2 = (V_{12}, E'_2)$. Заметим, что ориентированный подграф G^1 и орграф G_1

(ориентированный подграф G^2 и орграф G_2) изоморфны. Таким образом, степень любой вершины G_{12} является удвоенной степенью вершины с тем же номером орграфа G_1. Обозначим степенную последовательность ориентированного мультиграфа G_{12} - $d_2(n)$.

Выделим следующие свойства G_{12}.

Свойство 3. Для любой вершины мультиграфа G_{12} количество дуг, исходящих из нее и принадлежащих множеству E_1, равно количеству дуг, исходящих из нее и принадлежащих множеству E_2'; а количество дуг, входящих в нее и принадлежащих множеству E_1, равно количеству дуг, входящих в нее и принадлежащих множеству E_2'.

Свойство 4. G_{12} может иметь кратные дуги (кратности не выше 2), причем если e_1, e_2 — кратные дуги G_{12}, то $e_1 \in E_1, e_2 \in E_2'$ или $e_1 \in E_2', e_2 \in E_1$.

Рассмотрим некоторое подмножество $A_2(d_2(n))$ множества ориентированных мультиграфов $M_2(d_2(n))$ с занумерованными вершинами, обладающих степенной последовательностью $d_2(n)$ и удовлетворяющих свойствам 3 и 4. Определим операцию

$$\omega(A_2(d_2(n))) = \{G_{12}' |\ \exists\, G_{12} = (V_{12}, E_{12}) \in M_2(d_2(n)),$$
$$\exists\, v_1, v_2, v_3, v_4 \in V_{12},\ (v_1, v_2), (v_3, v_4) \in E_1,\ (v_1, v_4), (v_3, v_2) \notin E_1$$

(или $(v_1, v_2), (v_3, v_4) \in E_2',\ (v_1, v_4), (v_3, v_2) \notin E_2'$),

$$G_{12}' = (V_{12}, (E_1 \cup \{\, (v_1, v_4), (v_3, v_2)) \,\}) \setminus \{\, (v_1, v_2), (v_3, v_4) \,\})$$

(или $G_{12}' = (V_{12}, (E_2' \cup \{\, (v_1, v_4), (v_3, v_2) \,\}) \setminus \{\, (v_1, v_2), (v_3, v_4) \,\}))\}$.

Будем говорить, что операция ω эффективно применима в ориентированном мультиграфе G_{12} к некоторой паре дуг $(v_{i_1}, v_{i_2}) \in E_1$, $(v_{j_1}, v_{j_2}) \in E_1$ (или $(v_{i_1}, v_{i_2}) \in E_2', (v_{j_1}, v_{j_2}) \in E_2'$), если в G_{12} не существует дуг $(v_{i_1}, v_{j_2}) \in E_1, (v_{j_1}, v_{i_2}) \in E_1$ ($(v_{i_1}, v_{j_2}) \in E_2', (v_{j_1}, v_{i_2}) \in E_2'$), и при удалении дуг $(v_{i_1}, v_{i_2}) \in E_1, (v_{j_1}, v_{j_2}) \in E_1$ ($(v_{i_1}, v_{i_2}) \in E_2'$, $(v_{j_1}, v_{j_2}) \in E_2'$) и добавлении дуг $(v_{i_1}, v_{j_2} \in E_1, (v_{j_1}, v_{i_2}\} \in E_1$ ($(v_{i_1}, v_{j_2}) \in E_2', (v_{j_1}, v_{i_2}) \in E_2'$) в полученном ориентированном мультиграфе G_{12}' количество кратных дуг увеличивается.

Перед описанием алгоритма докажем следующие леммы.

Лемма 6. *Если существуют не кратные дуги* $(v_{j_2}, v_{j_1}) \in E_1$, $(v_{j_2}, v_{j_3}) \in E_2'$ *и кроме этих двух дуг, возможно, существует не* *кратная дуга* $(v_{j_1}, v_{j_3}) \in E_2'$, *либо пара кратных дуг* $(v_{j_1}, v_{j_3}) \in E_1$, $(v_{j_1}, v_{j_2}, v_{j_3}$ *— все различные вершины), то в ориентированном мультиграфе* $G_{12} = (V_{12}, E_{12})$ *существуют две дуги, к которым эффективно применима операция* ω.

Доказательство.

Шаг 1. По свойству 1 ориентированного мультиграфа G_{12} существует дуга $(v_{j_4}, v_{j_3}) \in E_1$.

Если существует дуга $(v_{j_4}, v_{j_1}) \in E_2'$, то к дугам $(v_{j_2}, v_{j_3}), (v_{j_4}, v_{j_1})$ применяем операцию ω. Получаем две пары кратных дуг (v_{j_2}, v_{j_1}), (v_{j_4}, v_{j_3}). Остановка.

Если не существует дуги $(v_{j_4}, v_{j_1}) \in E_1$, то к дугам (v_{j_2}, v_{j_1}), (v_{j_4}, v_{j_3}) применяем операцию ω. Получаем пару кратных дуг (v_{j_2}, v_{j_3}). Остановка.

Иначе существует дуга $(v_{j_4}, v_{j_1}) \in E_1$.

Переход к шагу 2.

Шаг 2. Так как исходящих из вершины v_{j_4} дуг, принадлежащих множеству E_1, больше, чем исходящих из вершины v_{j_4} дуг, принадлежащих множеству E_2', то по свойству 3 для вершины v_{j_4} существует исходящая дуга $(v_{j_4}, v_{j_5}) \in E_2'$.

Если существует дуга $(v_{j_2}, v_{j_5}) \in E_1$, то к дугам $(v_{j_2}, v_{j_5}), (v_{j_4}, v_{j_3})$ применяем операцию ω. Получаем две пары кратных дуг (v_{j_2}, v_{j_3}), (v_{j_4}, v_{j_5}). Остановка.

Если не существует дуги $(v_{j_2}, v_{j_5}) \in E_2'$, то к дугам (v_{j_2}, v_{j_3}), (v_{j_4}, v_{j_5}) применяем операцию ω. Получаем пару кратных дуг (v_{j_4}, v_{j_3}). Остановка.

Иначе существует дуга $(v_{j_2}, v_{j_5}) \in E_2'$.

Если существует дуга $(v_{j_1}, v_{j_5}) \in E_1$ и не существует дуги $(v_{j_1}, v_{j_3}) \in E_1$, то к дугам $(v_{j_1}, v_{j_5}), (v_{j_4}, v_{j_3})$ применяем операцию ω. Получаем пару кратных дуг (v_{j_4}, v_{j_5}). Остановка.

Если существует дуга $(v_{j_1}, v_{j_5}) \in E_1$ и дуга $(v_{j_1}, v_{j_3}) \in E_1$, значит существует дуга $(v_{j_1}, v_{j_3}) \in E_2'$. Далее, если не существует дуги $(v_{j_1}, v_{j_5}) \in E_2'$, то к дугам $(v_{j_1}, v_{j_3}), (v_{j_4}, v_{j_5})$ применяем операцию ω. Получаем пару кратных дуг (v_{j_4}, v_{j_3}). Остановка.

Иначе существует дуга $(v_{j_1}, v_{j_5}) \in E_2'$.

Переход к шагу 3.

Шаг $2k+1$. На предыдущем шаге получили ориентированный подграф $G_{2k+3} = (V_{2k+3}, E_{2k+3})$ ориентированного мультиграфа G_{12} такой, что $V_{2k+3} = \left\{ v_{j_1}, \ldots, v_{j_{2k+3}} \right\}$, а множество E_{2k+3} состоит,

во-первых, из не кратных дуг вида: $(v_{j_{2m+2}}, v_{j_{2l-1}}) \in E_1$, где $m = \overline{1,k}$, $l = \overline{1,m+1}$, и $\{v_{j_{2l}}, v_{j_{2m+1}}\} \in E_2'$, где $m = \overline{1,k+1}$, $l = \overline{1,m}$,

во-вторых, из некоторого подмножества не кратных дуг вида: $(v_{j_1}, v_{j_{2m+1}}) \in E_2'$, $m \leqslant k$, либо кратных дуг такого вида $(v_{j_1}, v_{j_{2m+1}})$ (но таким образом, что сохраняется свойство 4 ориентированного мультиграфа G_{12}),

и, в-третьих, из некоторого подмножества кратных или не кратных дуг, не нарушающих не кратности дуг, описанных выше.

При этом количество дуг из E_2', входящих в вершину $v_{j_{2k+3}}$, по крайней мере на 1 больше количества таких дуг из E_1. Отсюда вытекают следующие рассуждения.

Существует не кратная дуга $(v_{j_{2k+4}}, v_{j_{2k+3}}) \in E_1$, (следовательно, $j_{2k+4} \notin \{j_1, \ldots, j_{2k+2}\}$).

Далее для каждого $i = 2l+1$, $l = \overline{0,k}$ проверяем следующие два условия.

Если существует дуга $(v_{j_{2k+4}}, v_{j_i}) \in E_2'$, то к дугам $(v_{j_{2k+4}}, v_{j_i})$, $(v_{j_{2k+2}}, v_{j_{2k+3}})$ эффективно применима операция ω. Получаем две пары кратных дуг $(v_{j_{2k+2}}, v_{j_i}), (v_{j_{2k+4}}, v_{j_{2k+3}})$. Остановка.

Если не существует дуги $(v_{j_{2k+4}}, v_{j_i}) \in E_1$, то к дугам $(v_{j_{2k+2}}, v_{j_i})$, $(v_{j_{2k+4}}, v_{j_{2k+3}})$ эффективно применима операция ω. Получаем пару кратных дуг $(v_{j_{2k+2}}, v_{j_{2k+3}})$. Остановка.

Переход к шагу $2k+2$.

Шаг $2k+2$.

На предыдущем шаге получили ориентированный подграф $G_{2k+4} = (V_{2k+4}, E_{2k+4})$ ориентированного мультиграфа G_{12} такой, что $V_{2k+4} = \left\{ v_{j_1}, \ldots, v_{j_{2k+4}} \right\}$, а множество E_{2k+4} состоит,

во-первых, из не кратных дуг вида: $(v_{j_{2m+2}}, v_{j_{2l-1}}) \in E_1$, где $m = \overline{1,k+1}$, $l = \overline{1,m+1}$, и $\{v_{j_{2l}}, v_{j_{2m+1}}\} \in E_2'$, где $m = \overline{1,k+1}$, $l = \overline{1,m}$,

во-вторых, из некоторого подмножества не кратных дуг вида: $(v_{j_1}, v_{j_{2m+1}}) \in E_2'$, $m \leqslant k$, либо кратных дуг такого вида $(v_{j_1}, v_{j_{2m+1}})$ (но таким образом, что сохраняется свойство 4 ориентированного мультиграфа G_{12}),

и, в-третьих, из некоторого подмножества кратных или не кратных дуг, не нарушающих не кратности дуг, описанных выше.

При этом количество дуг из E_1, входящих в вершину $v_{j_{2k+4}}$, по крайней мере на 1 больше количества таких дуг из E_2' Отсюда вытекают следующие рассуждения.

Существует не кратная дуга $(v_{j_{2k+4}}, v_{j_{2k+5}}) \in E_2'$, (следовательно, $j_{2k+5} \notin \{j_1, \dots, j_{2k+3}\}$).

Далее для каждого $i = 2l,\ l = \overline{1, k+1}$ проверяем следующие два условия.

Если существует дуга $(v_{j_i}, v_{j_{2k+5}}) \in E_1$, то к дугам $(v_{j_i}, v_{j_{2k+5}})$, $(v_{j_{2k+4}}, v_{j_{2k+3}})$ эффективно применима операция ω. Получаем две пары кратных дуг $(v_{j_i}, v_{j_{2k+3}})$, $(v_{j_{2k+4}}, v_{j_{2k+5}})$. Остановка.

Если не существует дуги $(v_{j_i}, v_{j_{2k+5}}) \in E_2'$, то к дугам $(v_{j_i}, v_{j_{2k+5}})$, $(v_{j_{2k+4}}, v_{j_{2k+3}})$ эффективно применима операция ω. Получаем пару кратных дуг $(v_{j_{2k+4}}, v_{j_{2k+3}})$. Остановка.

Иначе существует дуга $(v_{j_i}, v_{j_{2k+5}}) \in E_2'$. Переходим к проверке для следующего i.

Если существует дуга $(v_{j_1}, v_{j_{2k+5}}) \in E_1$ и не существует дуги $(v_{j_1}, v_{j_{2k+3}}) \in E_1$, то к дугам $(v_{j_1}, v_{j_{2k+5}}), (v_{j_{2k+4}}, v_{j_{2k+3}})$ применяем операцию ω. Получаем пару кратных дуг $(v_{j_{2k+4}}, v_{j_{2k+5}})$. Остановка.

Если существует дуга $(v_{j_1}, v_{j_{2k+5}}) \in E_1$ и дуга $(v_{j_1}, v_{j_{2k+3}}) \in E_1$, значит существует дуга $(v_{j_1}, v_{j_{2k+3}}) \in E_2'$. Далее, если не существует дуги $(v_{j_1}, v_{j_{2k+5}}) \in E_2'$, то к дугам $(v_{j_1}, v_{j_{2k+3}}), (v_{j_{2k+4}}, v_{j_{2k+5}})$ применяем операцию ω. Получаем пару кратных дуг $(v_{j_{2k+4}}, v_{j_{2k+3}})$. Остановка.

Иначе существует дуга $(v_{j_1}, v_{j_{2k+5}}) \in E_2'$.

Далее полагаем $k := k + 1$ и переходим к шагу $2k + 1$.

Рассмотрим не кратные дуги $(v_{j_2}, v_{j_1}) \in E_1, (v_{j_2}, v_{j_3}) \in E_2'$, и, если таковые существуют, не кратную дугу $(v_{j_1}, v_{j_3}) \in E_2'$, либо пару кратных дуг $(v_{j_1}, v_{j_3}) \in E_1$, $(v_{j_1}, v_{j_2}, v_{j_3}$ — все различные вершины), применяем к ним последовательно шаги $1, 2$ и так далее. Допустим, остановка произойдет на шаге n. Это значит, в ориентированном мультиграфе G_{12} были найдены две дуги, к которым эффективно применима операция ω, в результате чего образуется пара кратных дуг, одна из которых принадлежит множеству E_1, а другая — множеству $E_2^{'}$) (или две пары кратных дуг). Иначе алгоритм находит бесконечную последовательность попарно различных подграфов ориентированного мультиграфа G_{12}, что невозможно в силу конечности G_{12}.

Лемма 6 доказана.

Аналогично доказывается Лемма 7.

Лемма 7. *Если существуют не кратные дуги* $(v_{j_2}, v_{j_1}) \in E_2'$, $(v_{j_2}, v_{j_3}) \in E_1$ *и кроме этих двух дуг, возможно, существует не кратная дуга* $(v_{j_1}, v_{j_3}) \in E_1$, *либо пара кратных дуг* $(v_{j_1}, v_{j_3}) \in E_2'$, *(*$v_{j_1}, v_{j_2}, v_{j_3}$ *— все различные вершины), то в ориентированном мультиграфе* $G_{12} = (V_{12}, E_{12})$ *существуют две дуги, к которым эффективно применима операция* ω.

Доказательство. Аналогично доказательству леммы 6.

Лемма 8. *Существует алгоритм, который для любой не кратной дуги ориентированного мультиграфа* G_{12} *с заданной нумерацией вершин строит простую цепь с началом в одной из вершин этой дуги, определяющую пару дуг, к которым эффективно применима операция* ω.

Доказательство. Рассмотрим произвольную не кратную дугу $(v_{j_2}, v_{j_1}) \in E_1$ (симметричный случай рассматривается аналогично). По свойству 3 существует не кратная дуга

$$(v_{j_2}, v_{j_3}) \in E_2',$$

а также существует не кратная дуга

$$(v_{j_4}, v_{j_3} \in E_1.$$

1) Если $j_4 = j_1$, рассмотрим следующие случаи:

если не существует не кратной дуги $(v_{j_3}, v_{j_1}) \in E_2'$, то к дугам (v_{j_2}, v_{j_3}), (v_{j_2}, v_{j_1}) применима Лемма 6(7); в результате найдем простую цепь

$$v_{j_3}, (v_{j_2}, v_{j_3}), v_{j_2}, (v_{j_2}, v_{j_1}), v_{j_1}, \ldots, v_{j_l},\ l \leqslant n$$

и пару дуг, к которым эффективно применима операция ω.

если существует не кратная дуга $(v_{j_3}, v_{j_1}) \in E_2'$ и не существует не кратной дуги $(v_{j_3}, v_{j_2}) \in E_1$, значит существует не кратная дуга $(v_{j_3}, v_{j_4}) \in E_1$, и к дугам (v_{j_2}, v_{j_1}), (v_{j_3}, v_{j_1}) применима Лемма 6(7) (при этом продолжением цепочки является уже найденная дуга (v_{j_3}, v_{j_4})); найдем простую цепь

$$v_{j_2}, (v_{j_2}, v_{j_1}), v_{j_1}, (v_{j_3}, v_{j_1}), v_{j_3}, (v_{j_3}, v_{j_4}), v_{j_4}, \ldots, v_{j_l},\ l \leqslant n$$

и пару дуг, к которым эффективно применима операция ω.

если существует не кратная дуга $(v_{j_3}, v_{j_1}) \in E_2'$ и существует не кратная дуга $(v_{j_3}, v_{j_2}) \in E_1$, значит

либо существует не кратная дуга $(v_{j_1}, v_{j_2}) \in E_2'$, и тогда, очевидно, полученный подграф ориентированного мультиграфа соответствует двум изоморфным подграфам исходных орграфов;

либо существует не кратная дуга $(v_{j_4}, v_{j_3}) \in E_2'$, значит к дугам (v_{j_3}, v_{j_1}), (v_{j_3}, v_{j_2}) применима Лемма 6(7) (при этом продолжением цепочки является уже найденная дуга (v_{j_4}, v_{j_3})); найдем простую цепь

$$v_{j_1}, (v_{j_3}, v_{j_1}), v_{j_3}, (v_{j_3}, v_{j_2}), v_{j_2}, (v_{j_4}, v_{j_2}), v_{j_4}, \ldots, v_{j_l},\ l \leqslant n$$

и пару дуг, к которым эффективно применима операция ω.

2) Если $j_4 \neq j_1$, то используя лемму 6(7), найдем простую цепь

$$v_{j_1}, (v_{j_2}, v_{j_1}), v_{j_2}, (v_{j_2}, v_{j_3}), v_{j_3}, (v_{j_4}, v_{j_3}), v_{j_4}, \ldots, v_{j_l},\ l \leqslant n$$

и пару дуг, к которым эффективно применима операция ω.

Лемма 8 доказана.

Замечание. Удлинение простой цепи в результате добавления дуги является результатом анализа окрестности вершины ориентированного мультиграфа, инцидентной данной дуге. Окрестность любой вершины ориентированного мультиграфа, состоящего из n вершин, содержит не более $n-1$ входящих или исходящих дуг, а простая цепь в таком мультиграфе состоит не более, чем из $n-1$ дуг, следовательно, алгоритм Леммы 8 заканчивает свою работу не более, чем за n^2 шагов.

Теперь будем преобразовывать ориентированный мультиграф $G_{12} = (V_{12}, E_{12})$, $|V_{12}| = n$, и соответствующие орграфы $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ с заданной нумерацией вершин в соответствии с Общим Алгоритмом.

Общий Алгоритм.
$G'_{12} := G_{12}$
$G'_1 := G_1, G'_2 := G_2$
Цикл:

Шаг 1. Если существует пара кратных дуг (v_i, v_j), то определим множество

$$E'_{12} := (E_1 \setminus (v_i, v_j)) \cup (E'_2 \setminus (v_i, v_j))$$

и перейдем к ориентированному мультиграфу $G'_{12} := (V_{12}, E'_{12})$. Орграфы G'_1, G'_2 остаются без изменений.

Если мультиграф G'_{12} пуст, то алгоритм заканчивает работу.

Переход к шагу 1.

Если в ориентированном мультиграфе G'_{12} не существует пары кратных дуг (v_i, v_j), то переход к шагу 2.

Шаг 2. Так как мультиграф G'_{12} непуст, существует дуга

$$(v_i, v_j) \in E_1 \ ((v_i, v_j) \in E'_2),$$

не являющяяся кратной. По Лемме 8 найдем пару дуг, к которым эффективно применима операция ω и применим ее в мультиграфе G'_{12} и к соответствующим дугам в орграфе G'_1 (G'_2). Обозначим полученные мультиграф и пару орграфов, соответственно, G'_{12} и G'_1 (G'_2).

Переход к шагу 1.

Для Общего Алгоритма справедлива следующая теорема.

Теорема 7. *Общий Алгоритм по данной паре орграфов* $\{G_1, G_2\} \in \Gamma^2$ *строит изоморфные орграфы* G'_1 *и* G'_2.

Доказательство. При работе алгоритма после каждого шага 2 во вновь полученном ориентированном мультиграфе возрастает количество кратных дуг, а общее число дуг не увеличивается. Поэтому после выполнения каждого цикла алгоритма (кроме, возможно, первого) число дуг мультиграфа G'_{12} уменьшается не менее, чем на 2. Следовательно, алгоритм работает конечное число шагов. В результате получим: $G'_{12} = (V_{12}, E'_{12}) \in \omega^N(\{G_{12}\})$ для некоторого N; все дуги мультиграфа G'_{12} — удалены; соответствующие орграфы

$$G'_1 \in \omega^{N_1}(\{G_1\})$$

и

$$G'_2 \in \omega^{N_2}(\{G_2\}).$$

Причем, при очередном $i+1$ применении операции ω к ориентированному мультиграфу из множества $\omega^i(\{G_{12}\})$ происходит применение операции ω к соответствующему орграфу из множества $\omega^i(\{G_1\})$ или из множества $\omega^i(\{G_2\})$, то есть $N_1+N_2=N$. Так как в результате работы алгоритма множество дуг мультиграфа G'_{12} пусто, то для каждой пары вершин v_i, v_j дуга (v_i, v_j) либо существует одновременно в графах G'_1, G'_2 и была удалена из мультиграфа G'_{12} на некотором шаге работы алгоритма, либо такой дуги не существует в орграфах G'_1, G'_2. Из этого следует, что G'_1 и G'_2 изоморфны.

Теорема 7 доказана.

Следствие 2. *Существуют целое неотрицательное число N и орграф G'_1 такие, что орграфы G'_1,*

$$G_1^{'} \in \omega^N(\{G_1\}) = \underbrace{\omega(\omega(\ldots\omega(\{G_1\})))}_{N}$$

и G_2 изоморфны (при $N=0 \quad \omega^N(\{G_1\}) = \{G_1\}$).

Далее рассмотрим некоторые характеристики изложенного алгоритма.

Под временной сложностью работы алгоритма будем подразумевать количество просмотренных за время его работы дуг; а под степенью вершины — общее количество инцидентных ей дуг.

Теорема 8. *Общий Алгоритм обладает следующими свойствами:*

1) *Временная сложность работы Общего Алгоритма на орграфе G с n вершинами, растет по порядку не быстрее, чем n^4.*
2) *Временная сложность работы Общего Алгоритма на орграфе G с n вершинами, степень каждой из которых не больше k, по порядку не превосходит k^2n^2.*
3) *Временная сложность работы Общего Алгоритма, обрабатывающего орграфы с n вершинами, степени которых не превосходят наперед заданной константы K, растет по порядку не быстрее, чем n^2.*

Доказательство. Следует из доказательства аналогичной теоремы для неориентированных графов.

3.3. Построение конечно-автоматного алгоритма для ориентированных графов

Рассмотрим вопрос о существовании конечного автомата A [1], который по паре $(G_1, G_2) \in \Gamma^2$ строил бы пару (G'_1, G'_2) такую, что $G'_i \in \omega^{N_i}(\{G_i\}), i = 1, 2$, и $G'_1 \cong G'_2$.

Определение 17. Конечно-автоматный алгоритм — это пара (A, T), где A — конечный автомат, T — некоторая двумерная таблица, каждый элемент t_{ij} которой является элементом некоторого конечного множества L. Автомат A применяется к таблице T в следующем смысле. В момент времени t автомат находится в некотором состоянии q_t. На вход автомата поступает некоторый элемент $t_{ij} \in T$. Выходом является элемент из L. Он записывается в эту клетку; автомат перемещается в одну из клеток с координатами $(i-1, j), (i, j-1), (i+1, j), (i, j+1)$ (при условии, что такая клетка существует) или остается на месте в клетке с координатами (i, j) и переходит в некоторое состояние q_{t+1}. В начальном состоянии q_0 автомат находится в клетке с координатами $(1, 1)$, и на вход автомату поступает элемент t_{11}. Работа автомата заканчивается, если он переходит в заключительное состояние q'.

Конечно-автоматный алгоритм работает по схеме: кодирование, обработка результатов кодирования некоторым конечным автоматом, декодирование.

Рассмотрим множество семерок

$$L = \{\{c_1, c_2, I_1^{out}, I_2^{out}, I_1^{in}, I_2^{in}, I\},$$
$$c_1 \in \{0, 1, 2, 3, 4, 5\},\ c_2 \in \{0, 1, 2, 3, 4, 5\},$$
$$I_1^{out} \in \{0, 1\}, I_2^{out} \in \{0, 1\}, I_1^{in} \in \{0, 1\}, I_2^{in} \in \{0, 1\}, I \in \{0, 1, 2\}\}.$$

Блок кодирования конечно-автоматного алгоритма паре орграфов $(G_1, G_2) \in \Gamma^2, G_i = (V_i, E_i),\ |V_i| = n, i = 1, 2$, взаимно однозначно сопоставляет таблицу специального вида — треугольник смежностей T размера n.

Определение 18. Треугольник смежностей размера n, сопоставленный паре графов $(G_1, G_2) \in \Gamma^2$, — это таблица с n строчками и n столбцами, $T = (t_{ij}), i = 1, 2, \ldots, n, j = i, i+1, \ldots, n$, каждый элемент t_{ij} которой является элементом конечного множества L.

Треугольник смежностей T взаимно однозначно сопоставляется паре графов $(G_1, G_2) \in \Gamma^2$ следующим образом: для $i \leqslant j$ положим

$$t_{ij} = \{c_1, c_2, I_1^{out}, I_2^{out}, I_1^{in}, I_2^{in}, I\},$$

где

$c_1 = c_2 = 0$,

$I_k^{out} = 1$, если дуга $(v_i, v_j) \in E_k$; $I_k^{out} = 0$, — иначе, $k = 1, 2$;

$I_k^{in} = 1$, если дуга $(v_j, v_i) \in E_k$; $I_k^{in} = 0$, — иначе, $k = 1, 2$;

$I = 1$, если $i = j,\ i \neq 1, n,\ i \neq 1, n$, либо $i = 1, 1 < j < n$, либо $j = n, 1 < i < n$; $I = 2$, если $i = j = 1,\ i = j = n,\ i = 1, j = n$; $I = 0$ — иначе.

Автомат A перерабатывает треугольник смежностей T, в соответствии с некоторой схемой, реализующей описанный в предыдущем пункте алгоритм, в треугольник смежностей T', по которому взаимно однозначно восстанавливается пара орграфов (G_1', G_2') такая, что $G_i' \in \omega^{N_i}(\{G_i\}), i = 1, 2$, и $G_1' \cong G_2'$.

Блок декодирования строит по T' такую пару (G_1', G_2') следующим образом: если

$$t_{ij} = \{c_1, c_2, I_1^{out}, I_2^{out}, I_1^{in}, I_2^{in}, I\},$$

то при $I_k^{out} = 1 \quad (v_i, v_j) \in E_k,\ k = 1, 2$, иначе — $(v_i, v_j) \notin E_k$, при $I_k^{in} = 1 \quad (v_j, v_i) \in E_k,\ k = 1, 2$, иначе — $(v_j, v_i) \notin E_k$.

Теорема 9. *Общий алгоритм, который по паре $(G_1, G_2) \in \Gamma^2$ строит пару (G_1', G_2') такую, что $G_i' \in \omega^{N_i}(\{G_i\}), i = 1, 2$, и $G_1' \cong G_2'$, является конечно-автоматным.*

Доказательство. Следует из доказательства аналогичной теоремы для неориентированных графов.

Общий Алгоритм является конечно-автоматным, поэтому размер задачи определим как количество ячеек треугольника смежностей, необходимого для задания входных данных алгоритма.

Теорема 10. *Объем памяти, необходимый для работы Общего Алгоритма, растет линейно относительно увеличения размера задачи.*

Доказательство. Следует из доказательства аналогичной теоремы для неориентированных графов.

4. Ограниченный по памяти алгоритм преобразования гиперграфов с заданной степенной последовательностью

4.1. Основные понятия

Определение 19 ([3]). Гиперграфом $H = (V, E)$ называется объект, состоящий из двух конечных множеств: V — называемого множеством вершин, и множества E непустых подмножеств элементов из V, называемого множеством гиперребер.

Определение 20 ([3]). Степенью вершины v называется число $d(v)$ гиперребер e_i, инцидентных этой вершине (то есть число таких ребер e_i, что $v \in e_i$).

Определение 21. Степенной последовательностью вершин гиперграфа $H = (V, E)$ назовем невозрастающую последовательность степеней всех вершин гиперграфа $H = (V, E)$. Обозначим ее $d_1(n)$.

Пусть даны два гиперграфа $H_1 = (V_1, E_1)$ и $H_2 = (V_2, E_2)$, причем $|V_1| = |V_2|$, $|E_1| = |E_2|$. Пусть существует взаимно однозначное отображение $f : V_1 \leftrightarrow V_2$ такое, что для любой вершины $v \in V_1$ выполняется условие: $v \in e, e \in E_1 \Leftrightarrow f(v) \in f(e), f(e) \in E_2$. Такое отображение называется изоморфизмом гиперграфов [3] $H_1 = (V_1, E_1)$ и $H_2 = (V_2, E_2)$. Если для графов H_1 и H_2 существует изоморфизм f, то H_1 и H_2 называются изоморфными (обозначение $H_1 \cong H_2$).

Рассмотрим некоторое подмножество $A_1(d_1(n))$ множества гиперграфов $M_1(d_1(n))$ с занумерованными вершинами, обладающих заданной степенной последовательностью вершин $d_1(n)$. Определим операцию

$$\begin{aligned}
\omega(A_1(d_1(n))) = \{H' |\ & \exists\, H = (V, E) \in A_1(d_1(n)), \\
& \exists\, v_1, v_2 \in V, e_1 \in E, e_2 \in E,\ v_1 \in e_1, v_2 \in e_2, v_1 \notin e_2, v_2 \notin e_1, \\
& H' = (V, (E \cup \{\, \{e_1 \setminus v_1\} \cup v_2, \{e_2 \setminus v_2\} \cup v_1 \,\}) \setminus \{\, e_1 \cup e_2 \,\})\}.
\end{aligned}$$

Очевидно, что операцию имеет смысл определять при $n \geqslant 4$. Кроме этого, отметим, что $\omega(A_1(d_1(n)))$ является подмножеством $M_1(d_1(n))$, то есть операция ω не меняет степенную последовательность вершин.

Будем говорить, что операция ω применима в гиперграфе H к некоторой паре гиперребер e_i, e_j, если в гиперграфе H существуют вершины $v_i,\ v_j$, такие что $v_i \in e_i,\ v_i \notin e_j$ и $v_j \in e_j,\ v_j \notin e_i$.

4.2. Постановка задачи. Описание алгоритма

Далее будем рассматривать множества гиперграфов с количеством вершин, не меньшим 4.

Зададим следующее взаимно однозначное отображение g из множества гиперграфов в некоторое подмножество ориентированных графов. Построим ориентированный граф $g(H) = H_g = (V_g, E_g)$ для гиперграфа $H = (V, E)$: пусть $|V| = m,\ |E| = n,\ V = \{ v_1, \ldots, v_m \}, E = \{ e_1, \ldots, e_n \}$, тогда $V_g = V \cup \{ x_1, \ldots, x_n \},\ E_g = E_1 \cup \ldots \cup E_n$, где $E_i = \bigsqcup\limits_{v_k \in e_i} \{x_i, v_k\}, 1 \leqslant i \leqslant n$. Заметим что для любой вершины v_k количество исходящих из нее ребер равно 0, а для любой вершины x_i количество входящих в нее ребер равно 0.

Лемма 9. *Отображение g сохраняет операцию ω, то есть $\omega(g(H)) = g(\omega(H))$.*

Доказательство. Рассмотрим вершины x_1, v_1, x_2, v_2 такие, что $(x_1, v_1),\ (x_2, v_2) \in E_g$, а $(x_1, v_2), (x_2, v_1) \notin E_g$. Для соответствующих гиперграфов это означает, что $e_1 \in E, e_2 \in E, v_1 \in e_1, v_2 \in e_2, v_1 \notin e_2$, $v_2 \notin e_1$. При этом если $H_g^{'} = \omega(g(H)),\ H_g^{'} = (V_g^{'}, E_g^{'})$, то $V_g^{'} = V_g$, а

$$E_g^{'} = (E_g \cup \{ (x_1, v_2), (x_2, v_1) \}) \setminus \{ (x_1, v_1), (x_2, v_2) \}).$$

Для гиперграфов: $\omega(g(H)) = (V', E'),\ V' = V$, а

$$E' = (E \cup \{(e_1 \setminus v_1) \cup v_2, (e_2 \setminus v_2) \cup v_1\}) \setminus \{ e_1, e_2 \}.$$

Следовательно, $\omega(\{g(H)\}) = g(\omega(\{H\}))$.

Лемма доказана.

Обозначим Γ^2 множество пар $\{H_1, H_2\}$ гиперграфов с занумерованными вершинами, у которых совпадают степенные последовательности вершин.

Рассмотрим пару гиперграфов $\{H_1, H_2\} \in \Gamma^2$,

$$H_1 = (V_1, E_1), H_2 = (V_2, E_2), |V_1| = |V_2| = n.$$

Сформулируем следующую теорему (аналогичную теореме для неориентированных графов).

Теорема 11. *Существуют числа* N_1 *и* N_2 *и найдутся гиперграфы* H_1' *и* G_2' *такие, что* $G_i' \in \omega^{N_i}(\{G_i\}) = \underbrace{\omega(\omega(\ldots\omega(\{G_i\})))}_{N_i}$, $i = 1, 2$, *и* G_1' *и* G_2' *изоморфны (считаем по определению* $\omega^0(M) = M$*).*

Доказательство. Построим ориентированные графы $H_g^1 = (V_g^1, E_g^1)$ и $H_g^2 = (V_g^2, E_g^2)$. По теореме 6 существуют числа N_1 и N_2 такие, что найдутся изоморфные ориентированные графы $H_g^{1'} \in \omega^{N_1}(\{H_g^1\}) = \underbrace{\omega(\omega(\ldots\omega(\{H_g^1\})))}_{N_1}$ и $H_g^{2'} = \omega^{N_2}(\{H_g^2\}) = \underbrace{\omega(\omega(\ldots\omega(\{H_g^2\})))}_{N_2}$. Отображение g сохраняет операцию ω, то есть $\omega(\{H_g^1)\}) = g(\omega(\{H_1\}))$ и $\omega(\{H_g^2)\}) = g(\omega(\{H_2\}))$. Следовательно, $\omega^{N_1}(\{H_g^1\}) = g(\omega^{N_1}(\{H_1\}))$ и $\omega^{N_2}(\{H_g^2\}) = g(\omega^{N_2}(\{H_2\}))$. Так как отображение g взаимно однозначно, то $\omega^{N_1}(\{H_1\}) = \omega^{N_2}(\{H_2\})$.

Теорема доказана.

Отсюда и из Теорем 7, 8 следуют Теоремы 12 и 13.

Теорема 12. *Существует конечно-автоматный алгоритм, который по паре гиперграфов* $(H_1, H_2) \in \Gamma^2$ *строит пару гиперграфов* (H_1', H_2') *такую, что* $H_i' \in \omega^{N_i}(\{H_i\}), i = 1, 2$, *и гиперграфы* H_1' *и* H_2' *изоморфны.*

Поскольку алгоритм Теоремы 12 применяется к ориентированным графам, специально построенным по данным гиперграфам, то временные характеристики этого алгоритма изменяются, исходя из того, что для гиперграфа с n вершинами будет построен ориентированный граф с количествами вершин и ребер, по порядку не превосходящими 2^n и $n2^n$, соответственно.

Под временной сложностью работы алгоритма на гиперграфах будем подразумевать количество просмотренных за время его работы дуг соответствующих орграфов.

Теорема 13. *Алгоритм на гиперграфах обладает следующими свойствами:*

1) *Временная сложность работы алгоритма на гиперграфе G с n вершинами (а, следовательно, на орграфе, содержащем по порядку не более, чем 2^n вершин и $n2^n$ ребер), растет по порядку не быстрее, чем $n^2 2^{2n}$.*

2) *Временная сложность работы алгоритма на гиперграфе G с n вершинами (а, следовательно, на орграфе с не более, чем 2^n вершинами), степень каждой из которых не больше k (следовательно, количество ребер орграфа по порядку не превосходит $k2^n$), по порядку не превосходит $k^2 2^{2n}$.*

3) *Временная сложность работы алгоритма на гиперграфе G с n вершинами (а, следовательно, на орграфе с не более, чем 2^n вершинами), степени которых не превосходят наперед заданной константы K, растет по порядку не быстрее, чем 2^{2n}.*

Алгоритм на гиперграфах является конечно-автоматным, поэтому размер задачи определим как количество ячеек треугольника смежностей, необходимого для задания входных данных алгоритма, то есть пары ориентированных графов с количеством вершин, по порядку не превосходящим 2^n.

Теорема 14. *Объем памяти, необходимый для работы алгоритма на гиперграфах, растет линейно относительно увеличения размера задачи.*

Доказательство. Следует из аналогичной теоремы для ориентированных графов.

5. Благодарности

Автор выражает глубокую признательность своему научному руководителю А. А. Часовских, неизменно поощрявшему работу на всех этапах, за постановку задач, постоянное внимание в ходе исследования, а также ценные замечания и предложения, существенно повлиявшие на теоретическую часть работы.

Список литературы

[1] Кудрявцев В. Б., Алешин С. В., Подколзин А. С. Введение в теорию автоматов. — М.: Наука, 1985.

[2] Яблонский С. В. Введение в дискретную математику. — М.: Наука, 1979.

[3] Емеличев В. А., Мельников О. И., Сарванов В. И. Тышкевич Р. И. Лекции по теории графов. — М.: Наука, 1990.