Serdica
Bulgariacae mathematicae
publicationes

Сердика
Българско математическо
списание

АЛГОРИТМ ДЛЯ НАХОЖДЕНИЯ ВСЕХ ИЗОМОРФИЗМОВ ДВУХ ГРАФОВ

АТАНАС А. РАДЕНСКИ

В работе предлагается эффективный алгоритм для нахождения всех изоморфизмов двух неориентированных или ориентированных графов. Время, необходимое для нахождения одного изоморфизма двух n-вершинных графов, пропорционально n^2. Алгоритм можно использовать, например, для решения некоторых задач химии и теории электрических цепей.

1. Введение. Пусть L — неориентированный связный униграф без петель с вершинами $x_1, x_2, \ldots, x_n$. Элементы матрицы смежности $A_L=(a_{ij})$ графа L определяются следующим образом:

если вершины x_i и x_j смежны, то $a_{ij}=a_{ji}=1$;

если вершины x_i и x_j несмежны, то $a_{ij}=a_{ji}=0$ $(i, j=1, 2, \ldots, n)$ Отметим, что вершина x_i несмежна с вершиной x_i. Матрица смежности A_L вполне определяет граф L.

Пусть L и M — два графа с матрицами смежности (a_{ij}) и (b_{ij}) и с множествами вершин $X\equiv\{x_1, x_2, \ldots, x_n\}$ и $Y\equiv\{y_1, y_2, \ldots, y_n\}$. Графы L и M называются изоморфными ($L\cong M$), если существует взаимно однозначное отображение φ множества X на множество Y, такое, что из равенств $y_p=\varphi(x_i)$, $y_q=\varphi(x_j)$ следует $a_{ij}=b_{pq}$ $(i, j=1, 2, \ldots, n)$. Теоретически проблема нахождения всех изоморфизмов графов L и M решается просто — проверкой всех $n!$ взаимно однозначных отображений φ множества X на Y. Однако практически такое решение неприменимо. Для некоторых специальных графов существуют эффективные решения проблемы изоморфизма [2, 3, 4]. В работе [1] предлагается общий алгоритм для нахождения всех изоморфизмов двух графов, у которого порядок быстродействия — n^5. Этот алгоритм основан на нахождении длинного списка характеристик. Совокупность списков предложенного вида оказывается инвариантной относительно изоморфизма. В [6] описан один из интереснейших алгоритмов для проверки изоморфности двух графов, в котором используется одно недоказанное предположение.

В настоящей работе предлагается весьма простая характеристика, определяющая граф с точностью до изоморфизма. Эта характеристика позволяет выяснить, изоморфен ли граф L графу M, и если они изоморфны, найти все изоморфизмы. Предложенный алгоритм без особых изменений применим и к случаю, когда L и M — произвольные униграфы.

2. Определения. Подграфом графа L называется каждый граф L' с множеством вершин X', $X'\subseteq X$, матрица смежности которого получена из A_L путем удаления всех строк и столбцов с номерами i, для которых $x_i\in X$ и $x_i\notin X'$ $(i=1, 2, \ldots, n)$, т. е. две вершины из X' смежны в L' тогда и только тогда, когда они смежны в L. Подграф L' графа L, порожденный множеством вершин X', будем обозначать иногда как $G_L(X')$.

Пусть $X' \subset X$, $X'' \subset X$, $X' \neq \emptyset$, $X'' \neq \emptyset$, $X' \cap X'' = \emptyset$. Комбинацией $G_L(X') * G_L(X'')$ подграфов $G_L(X')$ и $G_L(X'')$ будем называть граф с множеством вершин $X' \cup X''$, у которого сохранены все ребра из L, которые соединяют вершину из X' с вершиной из X''.

Количество вершин графа L будем обозначать через $|L|$, а множество его вершин — $V(L)$. Степенью вершины x_i относительно графа L будем называть число $S(x_i, L) = \Sigma a_{ij}$, где (a_{ij}) — матрица смежности графа L, т. е. число ребер, соединяющих вершину x_i с другими вершинами из L, есть $S(x_i, L)$. Заметим, что если L' — подграф графа L, то $S(x_i, L') \leq S(x_i, L)$.

Пусть $t(L) = \max\limits_{x_i \in V(L)} S(x_i, L)$. Подграфом степени r_1 будем называть подграф L^{r_1} графа L с вершинами все вершины из L, у которых степень есть r_1 ($r_1 = 0, 1, 2, \ldots, t(L)$). Рассматривая L^{r_1}, его подграф степени r_2 будем обозначать через $L^{r_1 r_2}$ $(r_2 = 0, 1, 2, \ldots, t(L^{r_1}))$ и т. д. Процесс получения новых подграфов данной степени приводит к тому, что получается разбиение графа $L : L_1, L_2, \ldots, L_{s_1}$, такое, что $S(x, L_i) = S(y, L_i)$ для любых вершин x и y графа L_i $(i = 1, 2, \ldots, s_1)$.

3. Некоторые необходимые условия изоморфизма двух графов. Пусть взаимно однозначное отображение φ определяет изоморфизм двух графов L и M.

Теорема 1. *Если x_i — произвольная вершина графа L, то $S(x_i, L) = S(\varphi(x_i), M)$.*

Доказательство. Пусть $y_p = \varphi(x_i)$. Так как $L \simeq M$, то $a_{ij} = b_{pq_j}$ $(j = 1, 2, \ldots, n)$, при этом $\{q_1, q_2, \ldots, q_n\} = \{1, 2, \ldots, n\}$. Следовательно,

$$S(x_i, L) = \sum_{j=1}^{n} a_{ij} = \sum_{j=1}^{n} b_{pq_j} = \sum_{k=1}^{n} b_{pk} = S(y_p, M) = S(\varphi(x_i), M).$$

Этот результат показывает, что если отображение φ определяет изоморфизм графов L и M, то степени образа и прообраза всегда одинаковы.

Лемма. *Пусть φ определяет изоморфизмы следующих графов: $L \simeq M$, $L' \simeq M'$, $L'' \simeq M''$, где L', L'' и M', M'' — подграфы графов L и M соответственно. Пусть выполняются условия для образования графов $L' * L''$ и $M' * M''$. Тогда $L' * L'' \simeq M' * M''$.*

Мы не будем доказывать лемму — легко показать, что φ определяет изоморфизм графов $L' * L''$ и $M' * M''$.

Теорема 2. *Если $(r_1, r_2, \ldots, r_k)$ — произвольный набор целых неотрицательных чисел, то $L^{r_1 r_2 \ldots r_k} \simeq M^{r_1 r_2 \ldots r_k}$.*

Доказательство. Теорема доказывается путем индукции по k. Пусть $k = 1$. Если $x \in V(L^{r_1})$ и $y \in V(M^{r_1})$, то $S(x, L) = S(y, M) = r_1$. Из теоремы 1 следует, что для каждой вершины $x \in V(L^{r_1})$ выполняется $\varphi(x) \in V(M^{r_1})$ и для каждой вершины $y \in V(M^{r_1})$ выполняется $\varphi^{-1}(y) \in V(L^{r_1})$. Следовательно, φ осуществляет однозначно обратимое отображение множества $V(L^{r_1})$ на $V(M^{r_1})$. Так как φ определяет изоморфизм графов L и M, то φ определяет изоморфизм графов L^{r_1} и M^{r_1}.

Пусть для каждого набора из $k=p-1$ целых неотрицательных чисел $(r_1, r_2, \ldots, r_{p-1})$ выполняется $L^{r_1r_2\ldots r_{p-1}} \cong M^{r_1r_2\ldots r_{p-1}}$. Пусть этот изоморфизм определятся однозначно обратимым отображением $\varphi_{r_1r_2\ldots r_{p-1}}$ множества $V(L^{r_1r_2\ldots r_{p-1}})$ на $V(M^{r_1r_2\ldots r_{p-1}})$. Тогда доказательство изоморфизма графов $L^{r_1r_2\ldots r_{p-1}r_p}$ и $M^{r_1r_2\ldots r_{p-1}r_p}$ проводится так же, как и доказательство изоморфизма L^{r_1} и M^{r_1} $(r_p=0, 1, 2, \ldots)$.

4. Основные этапы алгоритма для нахождения всех изоморфизмов двух графов. Неформальное описание алгоритма необходимо для выявления сути метода нахождения всех изоморфизмов. Опишем три основных этапа функционирования алгоритма. Пусть L и M — графы с множествами вершин $X=\{x_1, x_2, \ldots, x_n\}$ и $Y=\{y_1, y_2, \ldots, y_n\}$.

Этап 1. Этот этап состоит в получении разбиений графов L и M (или множеств $V(L)$ и $V(M)$, что одно и то же). Пусть $t(L)=t(M)$; если $t(L)\neq t(M)$, то из теоремы 1 следует, что L и M не могут быть изоморфными. Находим все графы L^{r_1}, $M^{r_1}(r_1=0, 1, 2, \ldots, t(L))$. Графы $L^0, L^1, \ldots, L^{t(L)}$ являются разбиением графа L. Потом находим все $L^{r_1r_2}$, $M^{r_1r_2}$, $L^{r_1r_2r_3}$, $M^{r_1r_2r_3}$ и т. д. — до тех пор, пока получаются все новые разбиения графов L и M. Если $L\cong M$, то из теоремы 2 следует, что для произвольного набора целых неотрицательных чисел $(r_1, r_2, \ldots, r_k)$ выполняется $L^{r_1r_2\ldots r_k}\cong M^{r_1r_2\ldots r_k}$. Если существует набор $(r_1, r_2, \ldots, r_k)$, для которого $|L^{r_1r_2\ldots r_k}|\neq|M^{r_1r_2\ldots r_k}|$, то L и M не могут быть изоморфными. В случае, когда не существует такого набора, определяем следующее соответствие: $V(L^0)\leftrightarrow V(M^0)$, $V(L^1)\leftrightarrow V(M^1), \ldots, V(L^{t(L)})\leftrightarrow V(M^{t(L)})$, потом $V(L^{ij})\leftrightarrow V(M^{ij})$ и т. д.

Результатом работы этапа 1 являются разбиения графов L и M. Если подграфы из разбиения графа L обозначим через $L_1, L_2, \ldots, L_{s_1}$, соответствующие им подграфы графа M — через $M_1, M_2, \ldots, M_{s_1}$, то все вершины графов L_i и M_i — одинаковой степени $S(i)$ $(1\leq i\leq s_1)$. Кроме того, выполняются следующие равенства:

$$\bigcup_{i=1}^{s_1} V(L_i)=X,\ \bigcup_{i=1}^{s_1} V(M_i)=Y,\ V(L_i)\cap V(L_j)=V(M_i)\cap V(M_j)=\varnothing$$

$$(i\neq j;\ i, j=1, 2, \ldots, s_1).$$

При этом если $L\cong M$, то $L_i\cong M_i\,(i=1, 2, \ldots, s_1)$.

Этап 2. Пусть результатом работы этапа 1 являются разбиения, $\{L_i\}_1^{s_1}$ и $\{M_i\}_1^{s_1}$. Если существует индекс i, такой, что $|L_i|\neq 1$, то образуем $L_i * L_j$ и $M_i * M_j\,(i\neq j)$. Пусть приложим этап 1 к $L_i * L_j$ и $M_i * M_j$. Таким образом получим разбиения этих графов. Тогда мы получим и новые разбиения графов L и M, заменив L_i, L_j и M_i, M_j полученными разбиениями. Точнее, пусть применением этапа 1 получено разбиение $L^{(1)}, L^{(2)}, \ldots, L^{(l)}$ графа $L_i * L_j$ и разбиение $M^{(1)}, M^{(2)}, \ldots, M^{(l)}$ графа $M_i * M_j$. Если существует целое число m, $1\leq m\leq l$, для которого выполняется $|L^{(m)}|\neq|M^{(m)}|$, то L и M не могут быть изоморфными. С другой стороны, из $L\cong M$ следует $L_i * L_j\cong M_i * M_j$ и отсюда $L^{(m)}\cong M^{(m)}$ $(m=1, 2, \ldots, l)$. В случае $|L^{(m)}|=|M^{(m)}|$ $(m=1, 2, \ldots, l)$ поступаем следующим образом. В разбиении $L_1, L_2, \ldots, L_{s_1}$ графа L заменяем графы L_i и L_j подграфами графа L, у которых множества вершин — все непустые из множеств $V(L_i)\cap V(L^{(p)})$, $V(L_j)\cap V(L^{(p)})(p=1, 2, \ldots, l)$. Новое разбиение графа M получается таким же способом.

Пусть получились новые, более „мелкие“ разбиения графа L и графа $M: L_1, L_2, \ldots, L_{s_2}$ и $M_1, M_2, \ldots, M_{s_2}$. К этим разбиениям снова применяем этап 1 и этап 2, и т. д. до тех пор, пока больше не получаются более мелкие разбиения. Заметим, что при выполнении этапов 1 и 2 возможно

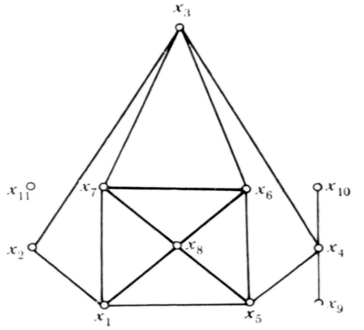

Рис. 1

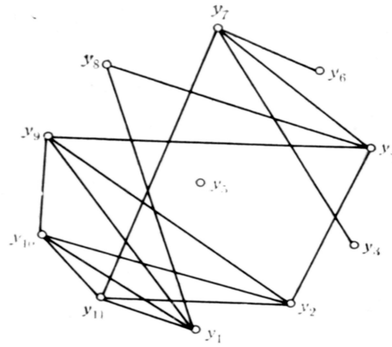

Рис. 2

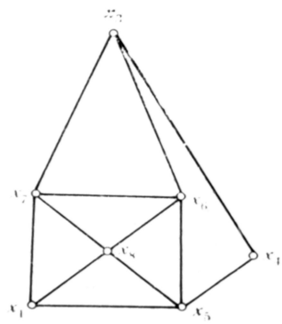

Рис. 3

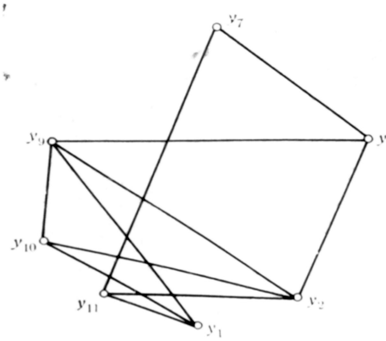

Рис. 4

получить одновершинные разбиения графов L и M, т. е. разбиения, в которых каждый подграф имеет только одну вершину. Тогда, как покажем далее, пары соответственных подграфов из этих разбиений определяют изоморфное соответствие вершин графов L и M.

Поясним на примере этапа 1 и этапа 2. Рассматриваем графы из рис. 1 и 2. Применяя этап 1, получаем пары соответственных множеств вершин:

$\{x_{11}\} \leftrightarrow \{y_5\}$ — вершины нулевой степени,

$\{x_9, x_{10}\} \leftrightarrow \{y_3, y_6\}$ — вершины первой степени,

$\{x_2\} \leftrightarrow \{y_8\}$ — вершины второй степени,

$\{x_1, x_3, x_4, x_5, x_6, x_7, x_8\} \leftrightarrow \{y_1, y_2, y_4, y_7, y_9, y_{10}, y_{11}\}$ — вершины четвертой степени. Из подграфов, порожденных множествами вершин четвертой степени (рис. 3 и 4) применением этапа 1 получаем разбиения последних двух множеств:

$\{x_4\}\leftrightarrow\{y_7\}$ — вершины второй степени,

$\{x_1, x_3\}\leftrightarrow\{y_1, y_4\}$ — вершины третьей степени,

$\{x_5, x_6, x_7, x_8\}\leftrightarrow\{y_2, y_9, y_{10}, y_{11}\}$ — вершины четвертой степени. Путем разбиения подграфов с множествами вершин $\{x_5, x_6, x_7, x_8\}$ и $\{y_2, y_9, y_{10}, y_{11}\}$

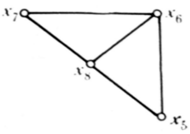

Рис. 5

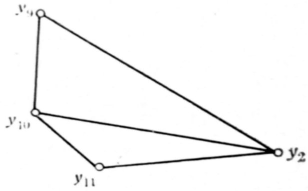

Рис. 6

(рис. 5 и 6) получаются новые пары соответственных множеств:

$\{x_5, x_7\}\leftrightarrow\{y_9, y_{11}\}$ — вершины второй степени,

$\{x_6, x_8\}\leftrightarrow\{y_2, y_{10}\}$ — вершины третьей степени.

Дальнейшие попытки получить новые разбиения этапом 1 оказываются безуспешными. Так, применением этапа 1 мы получаем следующие разбиения:

$$\{x_{11}\}, \{x_9, x_{10}\}, \{x_2\}, \{x_4\}, \{x_1, x_3\}, \{x_5, x_7\}, \{x_6, x_8\};$$

$$\{y_5\}, \{y_3, y_6\}, \{y_8\}, \{y_7\}, \{y_1, y_4\}, \{y_9, y_{11}\}, \{y_2, y_{10}\}.$$

К этим разбиениям применяем этап 2. Первое более мелкое разбиение находим из $G_L(\{x_4\}) * G_L(\{x_1, x_3\})$ и $G_M(\{y_7\}) * G_M(\{y_1, y_4\})$. Получаются пары соответственных множеств: $\{x_1\}\leftrightarrow\{y_1\}$ и $\{x_3, x_4\}\leftrightarrow\{y_4, y_7\}$. Так как уже имеется пара $\{x_4\}\leftrightarrow\{y_7\}$, то из $\{x_3, x_4\}\leftrightarrow\{y_4, y_7\}$ получаем $\{x_3\}\leftrightarrow\{y_4\}$. Графы $G_L(\{x_4\}) * G_L(\{x_5, x_7\})$ и $G_M(\{y_7\}) * G_M(\{y_9, y_{11}\})$ приводят к парам $\{x_5\}\leftrightarrow\{y_{11}\}$ и $\{x_7\}\leftrightarrow\{y_9\}$. Наконец, из комбинаций $G_L(\{x_1\}) * G_L(\{x_6, x_8\})$ и $G_M(\{y_1\}) * G_M(\{y_2, y_{10}\})$ находим $\{x_6\}\leftrightarrow\{y_2\}$ и $\{x_8\}\leftrightarrow\{y_{10}\}$. С помощью этапов 1 и 2 мы получили следующие разбиения:

$$\{x_{11}\}, \{x_9, x_{10}\}, \{x_2\}, \{x_4\}, \{x_1\}, \{x_3\}, \{x_5\}, \{x_7\}, \{x_6\}, \{x_8\};$$

$$\{y_5\}, \{y_3, y_6\}, \{y_8\}, \{y_7\}, \{y_1\}, \{y_4\}, \{y_{11}\}, \{y_9\}, \{y_2\}, \{y_{10}\}.$$

Этап 3. Пусть результатом работы этапов 1 и 2 являются разбиение $L_1, L_2, \ldots, L_s$ графа L и разбиение $M_1, M_2, \ldots, M_s$ графа M. В частности, возможно $s=1$, $L_1 \equiv L$ и $M_1 \equiv M$. Если $L \cong M$, то $L_i \cong M_i$ $(i=1, 2, \ldots, s)$. Мы уже отметили, что если $|L_i| = |M_i| = 1$ $(i=1, 2, \ldots, s)$, то соответствие $L_i\leftrightarrow M_i$ $(i=1, 2, \ldots, s)$ определяет изоморфное соответствие вершин графов L и M. Пусть существует пара графов L_i и M_i, у каждого из которых не менее двух вершин. В рассматриваемом примере такими являются L_2 и M_2 с множествами вершин $\{x_9, x_{10}\}$ и $\{y_3, y_6\}$. Этап 3 заключается в следующем: выбирается произвольная вершина $x_{(i)}$ графа L_i, ей ставится в сответствие произвольная вершина $y_{(i)}$ графа M_i. На месте старого разбиения графа L рассматривается новое: $L_1, L_2, \ldots, L_{i-1}, L'_i, L_{i+1}, \ldots, L_s, L_{s+1}$. Граф L'_i — подграф графа L с вершинами $V(L_i) - \{x_{(i)}\}$. У графа L_{s+1} только одна вершина — $x_{(i)}$. Таким же способом получается и новое разбиение графа $M: M_1, M_2, \ldots, M_{i-1}, M'_i, M_{i+1}, \ldots, M_s, M_{s+1}$. Граф M'_i — подграф графа M с вершинами $V(M_i) - \{y_{(i)}\}$. У графа M_{s+1} только одна вершина — $y_{(i)}$. Таким способом получаются новые, более мелкие разбиения графов L и M. Потом снова применяются этапы 1 и 2 до тех

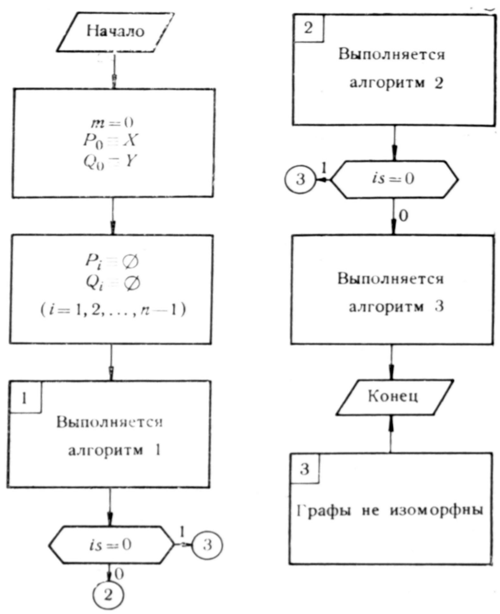

Рис. 7. Блок-схема АНВИ

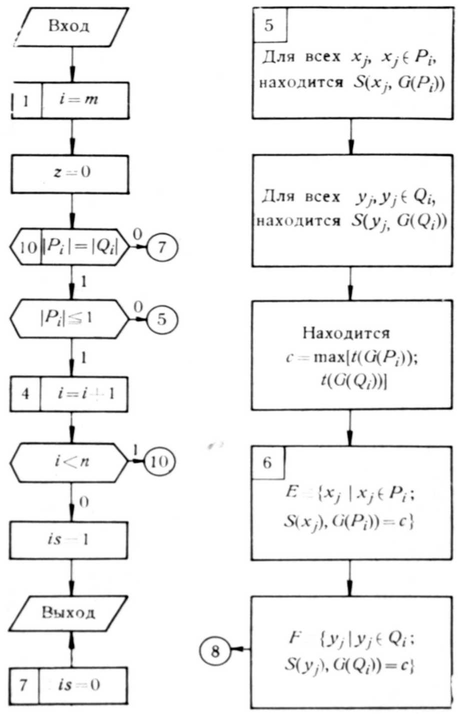

Рис. 8. Блок-схема алгоритма 1

пор, пока не будет необходимо использовать этап 3. Отметим, что если во время работы этапов 1 и 2 после применения этапа 3 получатся соответствующие подграфы L_i и M_i, для которых $|L_i| \neq |M_i|$, придется сделать шаг назад в этап 3 и сопоставить вершину $x_{(i)}$ с вершиной $y_{(i)} \neq y_{(i)}$.

Проделав все возможные сопоставления (во время работы этапа 3), получаем несколько однозначно обратимых отображений $\varphi_1, \varphi_2, \ldots, \varphi_k$ множества X на Y. Необходимо доказать следующие утверждения:

1) все φ_i $(i = 1, 2, \ldots, k)$ определяют изоморфизмы графов L и M.

2) если однозначно обратимое отображение φ определяет изоморфизм графов L и M, то существует номер z, такой, что $\varphi = \varphi_z$.

Доказательство этих утверждений проводится на основании блок-схемы алгоритма для нахождения всех изоморфизмов.

5. Алгоритм для нахождения всех изоморфизмов (АНВИ). Пусть требуется найти все изоморфизмы графов L и M с множествами вершин

$X \equiv \{x_1, x_2, \ldots, x_n\}$ и $Y \equiv \{y_1, y_2, \ldots, y_n\}$. Предложенные ниже алгоритмы (рис. 7—13) используют клетки P_i, P_i^j и Q_i, Q_i^j $(i=0, 1, \ldots, n-1;\ j=0, 1, \ldots, n)$, в которых записываются вершины графов L и M. Число вершин, записанных в P_i, обозначаем через $|P_i|$. Подграф графа L, порожденный вершинами из P_i, обозначаем через $G(P_i)$.

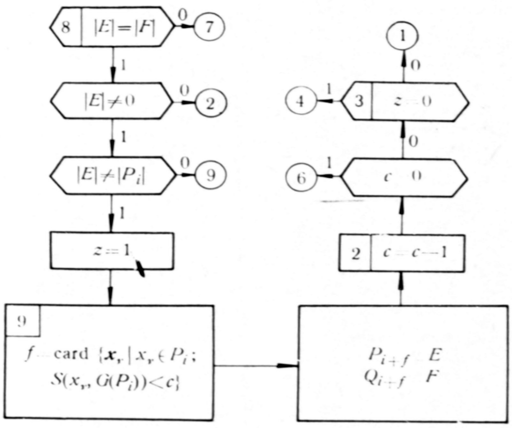

Рис. 9. Продолжение блок-схемы алгоритма 1

6. Две свойства АНВИ. Пусть L и M — два графа с множествами вершин $X \equiv \{x_1, x_2, \ldots, x_n\}$ и $Y \equiv \{y_1, y_2, \ldots, y_n\}$. Пусть результатом работы АНВИ над L и M являются однозначно обратимые отображения $\varphi_1, \varphi_2, \ldots, \varphi_k$ множества X на Y.

Теорема 3. *Если однозначно обратимое отображение φ множества X на Y определяет изоморфизм графов L и M, то существует номер z, такой, что $\varphi \equiv \varphi_z$.*

Доказательство. По окончании работы блоков АНВИ с этикетками 1 и 2 (рис. 7) вершины графа L записаны в клетках P_i, вершины графа M — в клетках Q_i $(i=0, 1, \ldots, n-1)$. Отметим, что в некоторых из этих клеток, возможно, ничего не записано. Следствием теорем 1 и 2 являются равенства $Q_i \equiv \varphi(P_i)$, $i=0, 1, \ldots, n-1$ (подразумевается, что $\varnothing \equiv \varphi(\varnothing)$). Очевидно, что если получено какое-либо разбиение вершин двух графов в клетках P_i и Q_i, для которого $Q_i \equiv \varphi(P_i)$ $(i=0, 1, \ldots, n-1)$, то после каждого применения алгоритмов 1 и 2 (т. е. этапов 1 и 2) к этим разбиениям получатся новые разбиения графов в клетках P_i и Q_i, для которых опять $Q_i \equiv \varphi(P_i)$ $(i=0, 1, \ldots, n-1)$.

В АНВИ после выполнения блоков с этикетками 1 и 2 следует выполнение алгоритма 3 (рис. 12—13). Пусть j — наименьший номер, такой, что $|P_j^1| > 1$. Пусть $x_f \in P_j^1$, $y_g \in Q_j^1$ и $y_g = \varphi(x_f)$. Во время выполнения алгоритма 3 для получения нового разбиения каждой вершине из P_j^1 всеми возможными способами ставится в соответствие вершина из Q_j^1. Тогда получится так, что вершина y_g будет сопоставлена с вершиной x_f. При этом сопоставлении вершины x_f и y_g записываются в клетках с одним и тем же номером, а вершины из $P_j^1-\{x_f\}$ и $Q_j^1-\{y_g\}$ — в других клетках, номера которых тоже одинаковы. Следовательно, опять будет выполнено равенство $Q_i\equiv\varphi(P_i)\,(i=0, 1, \ldots, n-1)$. Потом работа алгоритма 3 продол-

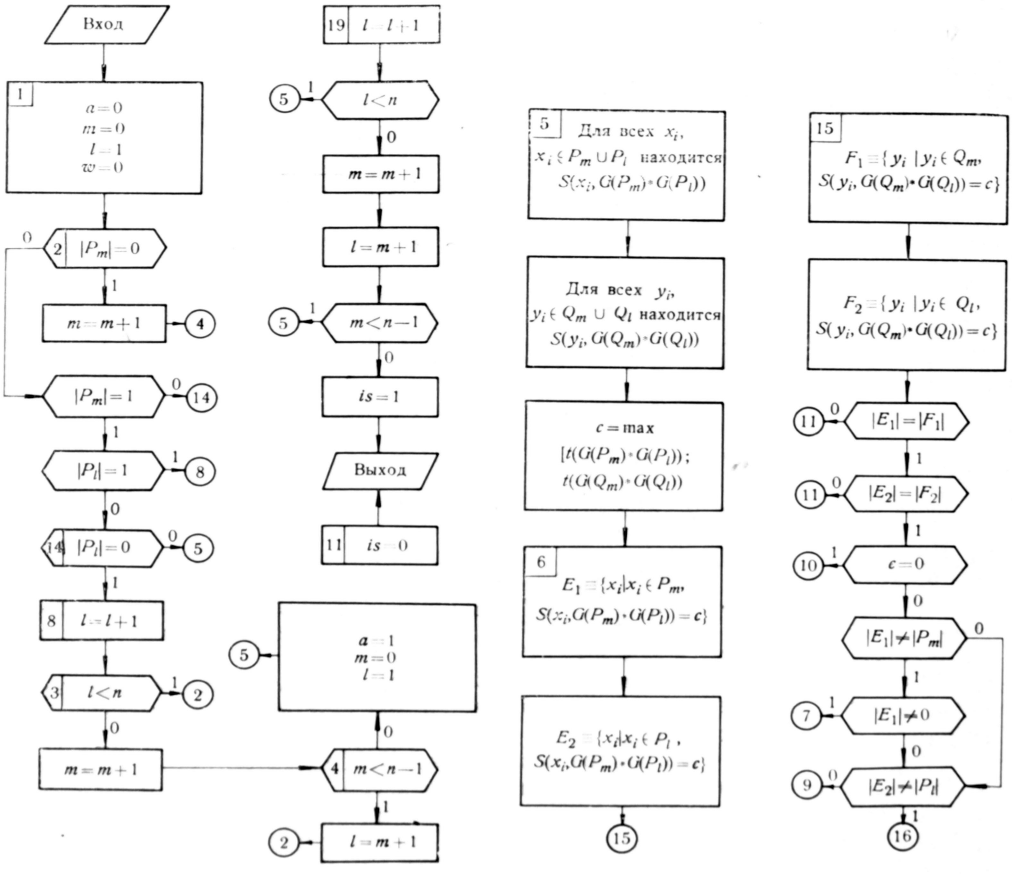

Рис. 10. Блок-схема алгоритма 2

жается выполнением алгоритмов 1 и 2 (см. блок с этикеткой 6), после которого $Q_i\equiv\varphi(P_i)$ $(i=0, 1, \ldots, n-1)$. Когда опять потребуется сопоставление двух вершин, оно будет сделано и после этого $Q_i\equiv\varphi(P_i)\,(i=0, 1,\ldots, n-1)$. Таким образом получатся одновершинные разбиения графов L и M, для которых $Q_i\equiv\varphi(P_i)$ $(i=0, 1,\ldots, n-1)$. Эти одновершинные разбиения определяют φ_z такое, что $\varphi\equiv\varphi_z$.

Теорема 4. *Пусть φ — произвольное взаимно однозначное отображение, полученное АНВИ. Тогда φ определяет изоморфизм L и M.*

Доказательство. Пусть x' и x'' — произвольные вершины графа L. Допустим, что x' и x'' смежные, а $\varphi(x')$ и $\varphi(x'')$ несмежные (аналогично рассматривается случай, когда x' и x'' несмежные, а $\varphi(x')$ и $\varphi(x'')$ — смежные). Отображение φ определяется некоторыми одновершинными раз-

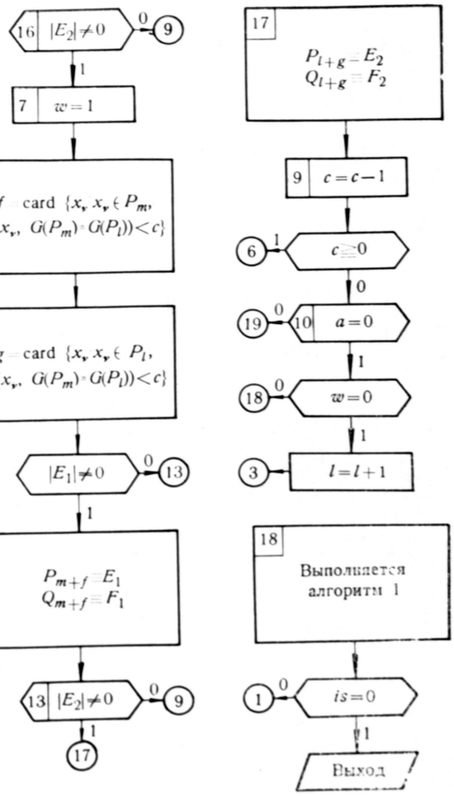

Рис. 11. Продолжение блок-схемы алгоритма 2

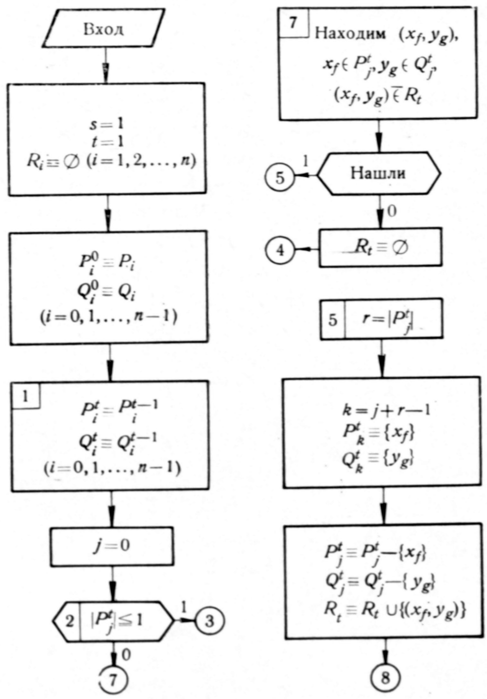

Рис. 12. Блок-схема алгоритма 3

биениями графов L и M. Следовательно, во время получения этих разбиений наступит момент, когда для некоторых $i, j: P_i = \{x'\}$, $Q_i = \{\varphi(x')\}$, $P_j = \{x''\}$, $Q_j = \{\varphi(x'')\}$. Рассмотрим выполнение Алгоритма 2, точнее, блок с этикеткой 5 и следующие два блока (рис. 10). Когда $m = \min(i, j)$, $l = \max(i, j)$, то $c = 1$, $E_1 = \{x'\}$, $F_1 = \varnothing$, следовательно $|E_1| \neq |F_1|$, и работа АНВИ прекращается без получения одновершинного разбиения, определяющего φ. Это противоречие является результатом того, что вершины x' и x'' смежны, а вершины, $\varphi(x')$ и $\varphi(x'')$ несмежны. Следовательно, x' и x'' смежны тогда и только тогда, когда смежны $\varphi(x')$ и $\varphi(x'')$, т. е. φ определяет изоморфизм графов L и M.

7. Заключение. Алгоритм для нахождения всех изоморфизмов можно использовать без принципиальных изменений и в более общем случае произвольных униграфов, у которых есть ребра N разных видов. В этом случае степенью вершины x будем называть N-мерный вектор $S(x)=(S_1, S_2, \ldots, S_N)$. Компонента S_i есть число ребер i-го вида, инцидентных с x.

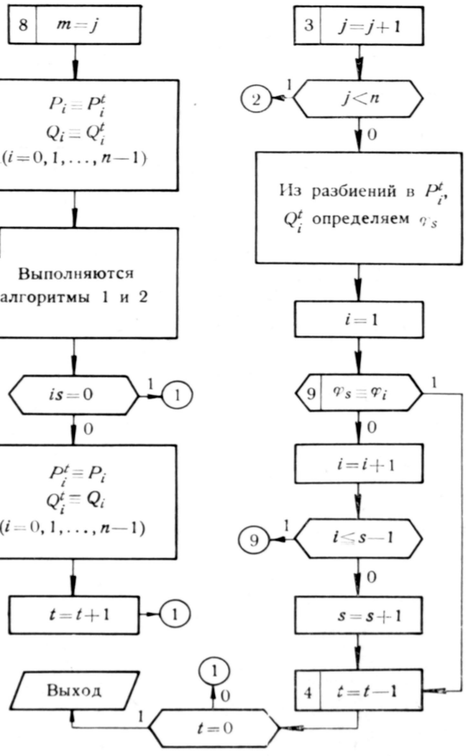

Рис. 13. Продолжение блок-схемы алгоритма 3

Были созданы программы для ЭВМ „Минск-32“, реализующие АНВИ. Почти все было запрограммированно на ФОРТРАНе (для болгарского транслятора ФЭР32), а остальное — на языке символического кодирования. Машинная реализация АНВИ позволила провести некоторые эксперименты для нахождения оценки скорости АНВИ (табл. 1).

Можно принять, что среднее время, необходимое для нахождения одного изоморфизма двух n-вершинных графов есть функция $t(n)$. Полином наилучшего среднеквадратичного приближения специального вида an^3+bn^2 для данных из таблицы является

$$t(n)=0{,}000048\,n^3+0{,}1225\,n^2.$$

Так как $0{,}000048 \ll 0{,}1225$, можно утверждать, что время работы АНВИ пропорционально n^2. Напомним, что время работы алгоритма из

Таблица 1

Результаты экспериментирования АНВИ

Количество вершин графов	Среднее время для нахождения одного изоморфизма (в s)	Количество исследованных пар графов	Среднее число ребер графов
10	9,4	9	22
20	39	12	128
30	119	10	310
40	276	9	539
70	523	3	483
100	1381	3	1901

[1] пропорционально n^5. Для сравнения можно использовать и результаты экспериментирования алгоритма из [5]. Программа, реализующая этот алгоритм, была использована не для нахождения изоморфного соответствия, а для установления факта, что два орграфа являются неизоморфными. Количество вершин графов, исследованных этим алгоритмом, не больше 12. Среднее время для установления неизоморфности двух 12-вершинных графов — 90 секунд. Сравнение этого времени с результатами АНВИ (табл. 1) выявляет преимущество АНВИ. Надо иметь в виду, что для экспериментов использована ЭВМ IBM 350/50, которая работает почти в 10 раз быстрее ЭВМ Минск-32. Можно ожидать, что оценка n^2 справедлива для всех графов, кроме сильнорегулярных (определение последних можно найти в [6]). Время работы алгоритма для сильнорегулярных графов будет повышаться в зависимости от степени регулярности графа.

Быстродействие АНВИ делает его удобным средством для решения практических задач. Возможно, этот подход позволит найти эффективный алгоритм для решения задачи изоморфного вхождения в n-мерный куб.

ЛИТЕРАТУРА

1. S. Franco. Isomorfismo fra grafi: un algoritmo effisiente per trovatte tutti gli isomorfismi. *Calcolo*, **8**, 1971, No 4, 301—337.
2. I. Hopcroft. A v^2 algorithm for determining isomorphism of planar graphs. *Inf. Process. Letters*, **1**, 1971, No. 1, 32—34.
3. L. Weinberg. A simple and efficient algorithm for determining isomorphism of planar triply connected graphs. *IEEE Trans. on Circuit Theory*, CT-13, 1966, 142—148.
4. Р. Шейнаускас. Алгоритм для установления изоморфизма и изоморфного вхождения двух графов. Выч. техника, т. 3. Каунас, 1972, 347—353.
5. A. Berztiss. A backtract procedure for isomorphism of directed graphs. *J ACM*, **20**, 1973, No 3.
6. D. Corneil, C. Gotlieb. An efficient algorithm for graph isomorphism. *J ACM*, **17**, 1970, No. 1, 51—64.

Единый центр науки и подготовки кадров по математике и механике
София 1000 п. я. 373

Поступила 20. 4. 1974