

нейронов; высокими требованиями к репрезентативности обучающего множества. Все это обуславливает необходимость наличия специальных навыков в использовании нейронных сетей при решении практических задач подобного класса.

СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Golovko, V. Neural Networks for Signal Processing in Measurement Analysis and Industrial Applications: the Case of Chaotic Signal Processing / V. Golovko, Y. Savitsky, N. Maniakov // Chapter of NATO book "Neural networks for instrumentation, measurement and related industrial applications". – Amsterdam: IOS Press, 2003. – P. 119–143.
2. Осовский, С. Нейронные сети для обработки информации / Пер. с польского И.Д. Рудинского. – М.: Финансы и статистика, 2002. – 334 с.
3. Golovko, V. Technique of Learning Rate Estimation for Efficient Training of MLP / V. Golovko, Yu. Savitsky, Th. Laopoulos, A. Sachenko, L. Grandinetti // Proc. of Int. Joint Conf. on Neural Networks IJCNN'2000, Como, Italy. – Vol. 1. – 2000. – P. 323–329.
4. Bezobrazova, S. Neural-network segmentation of electroencephalogram signal for epileptiform activity detection / S. Bezobrazova, V. Golovko // Computing. – 2008. – Vol 7, Issue 3 – P. 30–37.
5. Головкин, В.А. Нейросетевые методы определения спектра Ляпунова хаотических процессов / В.А. Головкин, Н.Ю. Чумерин // Нейрокомпьютеры: разработка и применение. – 2004. – № 1.
6. Временные сигналы ЭЭГ. Режим доступа: <http://www.meb.uni-bonn.de/epileptologie/science/physik/eeegdata.html>. – Дата доступа 15.10.2011.
7. Циммерман, Ф. Клиническая электрокардиография // Бином. – 2008.
8. Юрьева, О.Д. Исследование помехоустойчивости методов измерения длительности RR-интервала // Известия СПбГЭТУ «ЛЭТИ», Серия «Биотехнические системы в медицине и экологии». – СПб.: Изд-во СПбГЭТУ, 2007. – Вып. 1. – С. 9–19.
9. PhysioBank. – Mode of access: <http://www.physionet.org/> physio-bank/. – Date of access: 15.05.2012.
10. PhysioBank ATM, AHA Database [sample excluded record] (ahadb). – Mode of access: <http://www.physionet.org/cgi-bin/atm/ATM>. Dateofaccess: 15.05.2012.

Материал поступил в редакцию 13.11.13

SAVITSKY Y.V., DAVIDYUK J.I. Some Aspects of Applications of Neural Network Models for EEG and ECG Signals Analysis

Problems of applications of neural network technique for EEG and ECG signals analysis are considered. The solution of the task is based on original algorithm for Lyapunov exponent calculation. The analysis of experiments results on identification of abnormal zones in real EEG and ECG signals is made. The problem questions of this approach to biomedical signals processing are characterized.

УДК 004.272

Дунець Р.Б., Грига В.М.

ИССЛЕДОВАНИЕ МАТРИЧНЫХ МЕТОДОВ ПРЕДСТАВЛЕНИЯ ПОТОЧНЫХ ГРАФОВ АЛГОРИТМОВ

Введение. Различают следующие способы представления алгоритмов: вербально-дедуктивные, графические, матричные (табличные) и аналитические [1, 2].

Однако наиболее распространенными в вычислительной технике являются графические способы задания алгоритмов, которые широко используются для проектирования универсальных и специализированных вычислителей [3–6]. Преимуществами графических способов являются высокая наглядность непосредственного отражения структуры связей между элементами системы и широко описанные в теории графов методы их обработки. К основным недостаткам можно отнести трудности автоматической обработки графов в компьютерах и сложность отображения графов с большим количеством вершин [2].

Широкое применение теории поточных графов алгоритмов (ПГА) для проектирования специализированных вычислителей, в которой вершинам графов отвечают вычислительные операции (функциональные операторы), а дугам – линии передачи данных для обработки, позволяет проектировать, как правило, одноктаковые алгоритмические операционные устройства (ОАОП), имеющие высокие показатели быстродействия и значительные затраты оборудования [3–6]. Для достижения максимального быстродействия выполнения алгоритма ОАОП нужно конвейеризовать, что предполагает разбиение всего процесса вычисления на несколько уровней, но за счет добавления конвейерных регистров значительно возрастают аппаратные затраты.

К недостаткам поточных графов необходимо отнести бинарность всех операций, а также неразличение входов и выходов отдельных дуг (линий), которые поступают до одной вершины, не разрешая создавать иерархические вершины со многими входами и выходами. В вычислительной математике это имеет фундаментальное значение, поскольку $a - b \neq b - a$.

Поэтому для перехода от ПГА, которые разрешают строить одноктаковые алгоритмические операционные устройства к пространственно-временным графам алгоритмов [7, 8], которые дают возможность строить многоктаковые алгоритмические операционные устройства, необходимо предложить формальный подход, который бы позволял сохранять структуру ПГА в удобной для обработки форме и осуществлять переход к ее модифицированным вариантам [8], с целью исследования технических характеристик различных типов многоктаковых структур устройств. Для сохранения в компьютере структуры графа алгоритма и его обработки широко используются матрицы [2, 9, 10].

Матричные методы представления алгоритма в отличие от графических разрешают удобно представлять, обрабатывать и хранить в компьютере структуру алгоритма с произвольным количеством элементов. Существенным недостатком матричных методов задания структуры алгоритма является низкая наглядность [2].

Поскольку матричное представление ПГА значительно упрощает его хранение и обработку, а известные матричные методы не позволяют описывать вершины графа с входными и выходными портами, то в данной статье предложен матричный метод преобразования поточных графов, которые разрешают проектировать специализированные устройства для класса алгоритмов, чья структура не зависит от входных данных (алгоритмы без ветвлений или инвариантные к сдвигу алгоритмы [4, 6, 10]) с целью дальнейшего матричного перехода к основным вариантам структур пространственно-временных графов, позволяющих проектировать различные типы многоктаковых специализированных устройств.

Дунець Роман Богданович, д.т.н., профессор, заведующий кафедрой специализированных компьютерных систем Национального университета «Львовская политехника», г. Львов, Украина, e-mail: dunets@polynet.lviv.ua.

Грига Володимир Михайлович, аспирант кафедры специализированных компьютерных систем Национального университета «Львовская политехника», г. Львов, Украина, e-mail: vol_gr@mail.ru.

В основной части статьи предложен метод записи поточного графа алгоритма в форме матрицы связности портов и дуг, которая сохраняет структуру алгоритма в удобной для обработки форме и имеет линейную зависимость увеличения элементов матрицы от количества входных данных, а также рассмотрены особенности задания ПГА матрицами инциденций, матрицами смежности и структурной матрицей.

Предложенный метод записи ПГА в форме матрицы связности портов и дуг предусматривает выполнение следующих шагов:

- обозначения входных и выходных портов на ПГА;
- обозначения дуг, соединяющих вершины с помощью входных и выходных портов;
- процедуры записи информации.

На этапе обозначения входных и выходных портов ПГА осуществляют нумерацию входных и выходных портов для каждой вершины графа алгоритма слева направо числами от 1 до N , где N – количество натуральных чисел.

На этапе обозначения дуг, соединяющих вершины с помощью входных и выходных портов обозначают дуги записью $I_{N_j}^{N_i}$, где N_i – номер вершины, из исходного порта которой выходит дуга I , N_j – номер вершины, на входной порт которой входит дуга I .

На этапе записи ПГА осуществляют по правилу: в элемент матрицы с индексами $(P_{P_{in}}^{P_{out}}, I_{N_j}^{N_i})$ записывают единицу, если выходной порт P_{out} вершины N_j соединен с входным портом P_{in} вершины N_i , если соединений нет, записывается нуль.

Данная матрица связности портов и дуг имеет следующий вид:

$$Z = \begin{pmatrix} Z_{11} & Z_{12} & \dots & Z_{1l} \\ Z_{21} & Z_{22} & \dots & Z_{2l} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & Z_{ij} & \dots \\ \dots & \dots & \dots & \dots \\ Z_{p1} & Z_{p2} & \dots & Z_{pl} \end{pmatrix},$$

где $\forall Z_{ij} \in \{0,1\}$, $i = \overline{1, p}$, $j = \overline{1, l}$, p – количество пар входных и выходных портов, l – количество промежуточных дуг ПГА, соединяющих вершины с помощью пар входных и выходных портов.

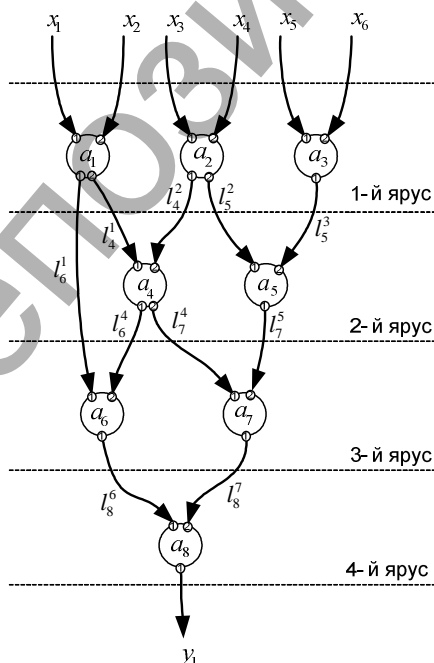


Рис. 1. Матричное преобразование ПГА с помощью матрицы связности портов и дуг

Матрица связности портов и дуг сохраняет следующую множество параметров $Z(P, C, F)$, где

$P = \{p_{ij}\}$ – множество пар входных и выходных портов, $i = \overline{1, p}$ – номер пары входных и выходных портов, p – общее количество пар входных и выходных портов;

$C = \{I_j\}$ – множество промежуточных дуг ПГА, $j = \overline{1, l}$ – номер промежуточной дуги ПГА, которая соединяет пары входных и выходных портов, l – количество промежуточных дуг ПГА, соединяющих соответствующие вершины с помощью пар входных и выходных портов;

$F = \{z_{ij}, F_z\} = \{0 \dots k\}$ – множество вершин ПГА, $k \in N$ (N – множество натуральных чисел), ij – номер вершины ПГА, $i = \overline{1, p}$, $j = \overline{1, l}$, F_z – общее количество вершин ПГА.

Информацию о выполняемых операциях каждым функциональным оператором (ФО) можно хранить в таблице операций, в которой каждому номеру ФО ставится в соответствие операция, выполняемая данным функциональным оператором.

На рис. 1 изображен произвольный ПГА с нумерацией входных и выходных портов и обозначением промежуточных дуг, который выполняет 8 однотопных операций, а также показан переход к матрице связности портов и дуг ($Z_{ПГА}$) данного поточного графа алгоритма.

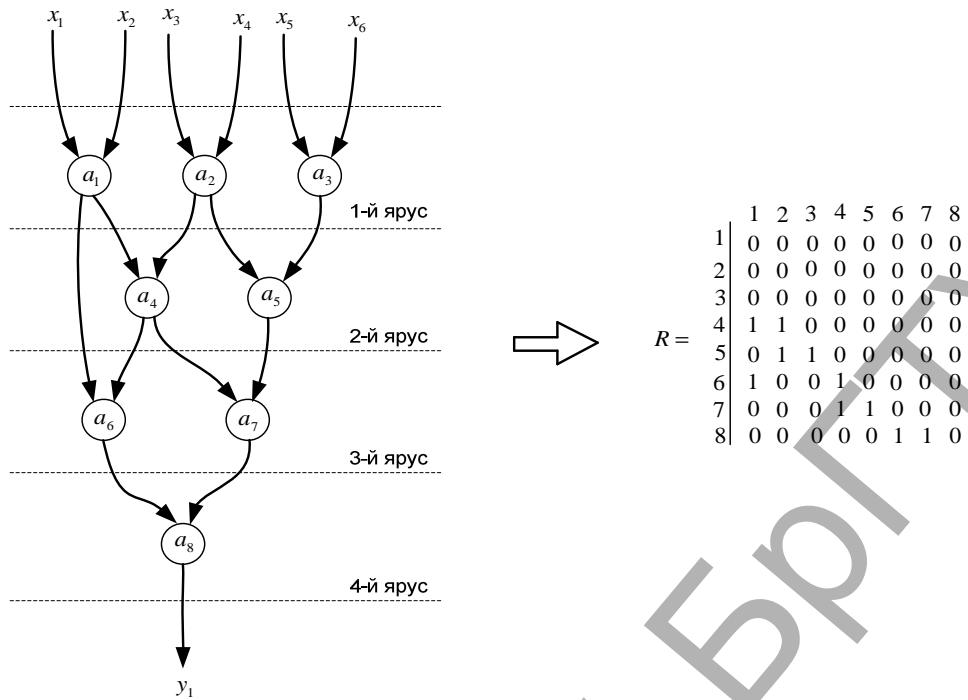
Общее количество элементов матрицы связности портов и дуг (S_{M3}) равно произведению пар входных и выходных портов (p) на количество дуг, которые их соединяют (d), поточного графа алгоритма:

$$S_{M3} = p \times d. \quad (1)$$

Размер образованной матрицы на рис. 1 составляет 30 элементов. Матрица связности портов и дуг разрешает получить множество пространственно-временных параметров ПГА, что дает возможность исследовать различные варианты преобразования ПГА путем сжатия его вершин и дуг, применяя пространственно-временные графы, и автоматизировать данный процесс.

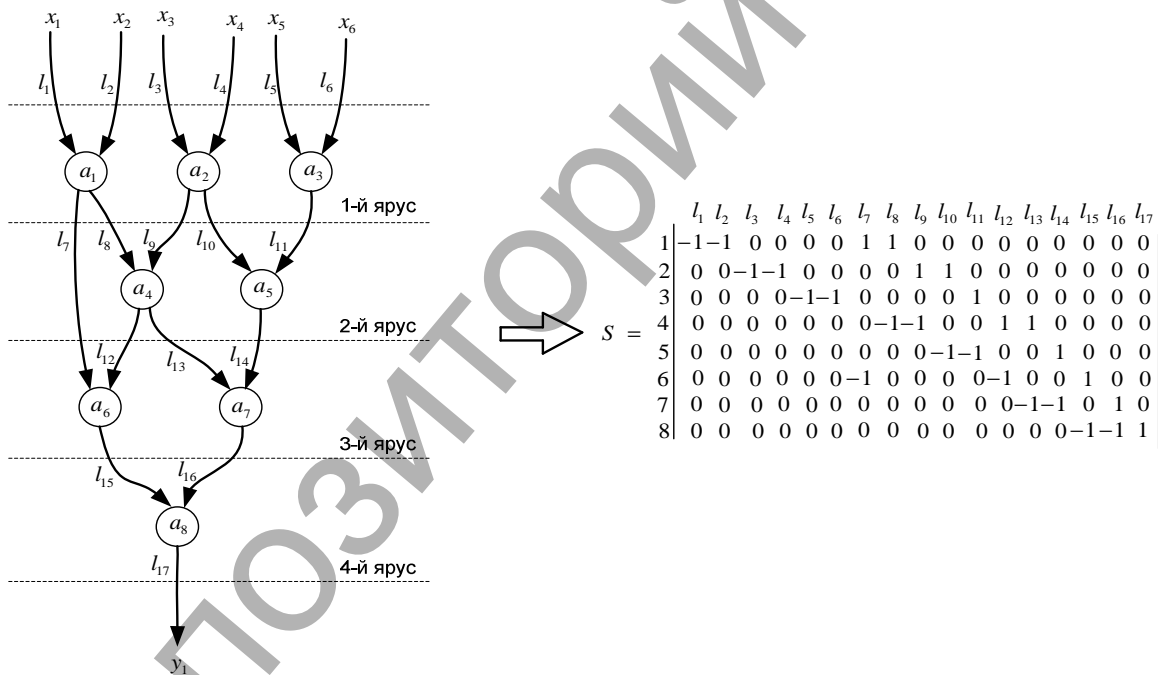
Матрица смежности – это квадратная матрица порядка N , где N – количество вершин графа [2, 9]. Заполнение матрицы смежности осуществляют по правилу: в элемент матрицы с индексами (i, j) ($i = \overline{1, N}$, $j = \overline{1, N}$) записывают количество дуг a ($a = \overline{1, n}$,

$$Z_{ПГА} = \begin{matrix} P_1^1 & \begin{vmatrix} I_6^1 & I_4^1 & I_2^1 & I_5^1 & I_3^1 & I_6^1 & I_4^1 & I_7^1 & I_5^1 & I_8^1 & I_7^1 \end{vmatrix} \\ P_1^2 & \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{vmatrix} \\ P_2^1 & \begin{vmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \end{vmatrix} \end{matrix}$$



$$R = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{matrix} & \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} \end{matrix}$$

Рис. 2. Матричное преобразование ПГА с помощью матрицы смежности



$$S = \begin{matrix} & \begin{matrix} l_1 & l_2 & l_3 & l_4 & l_5 & l_6 & l_7 & l_8 & l_9 & l_{10} & l_{11} & l_{12} & l_{13} & l_{14} & l_{15} & l_{16} & l_{17} \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{matrix} & \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 \end{bmatrix} \end{matrix}$$

Рис. 3. Матричное преобразование ПГА с помощью матрицы инцидентности

где n – количество входящих дуг графа), соединяющих вершины i и j , которые являются смежными, и вершина i является началом для данных дуг и нуль в других случаях.

На рис. 2 показан переход от ПГА к матрице смежности (R).

Общее количество элементов матрицы смежности (S_{MC}) равно квадрату количества вершин (N) поточного графа алгоритма:

$$S_{MC} = N^2. \quad (2)$$

Размер образованной матрицы на рис. 2 составляет 64 элемента.

Недостатком данного метода является то, что для представления графа алгоритма с помощью матрицы смежности общее количество вершин нужно знать заранее. Если граф должен создаваться и (или) изменяться, то для каждого добавления или уменьшения вер-

шины необходимо строить новую матрицу. Кроме того, если граф содержит небольшое количество дуг и матрица смежности состоит, в основном, из нулей, память должна быть выделена для всех возможных дуг, независимо от их существования.

Матрица инцидентности – это матрица размера $N \times M$, где N – количество вершин ПГА, M – количество дуг ПГА [2, 9]. Заполнение матрицы инцидентности осуществляют по правилу: в элемент матрицы с индексами (j, i) ($j = \overline{1, M}, i = \overline{1, N}$) записывают единицу, если вершина инцидентна дуге и является ее началом, в элемент (j, i) – минус единицу, если вершина инцидентна дуге и является ее концом, и нуль, если вершина и дуга не являются инцидентны.

На рис. 3 показан переход от ПГА к матрице инцидентности (S).

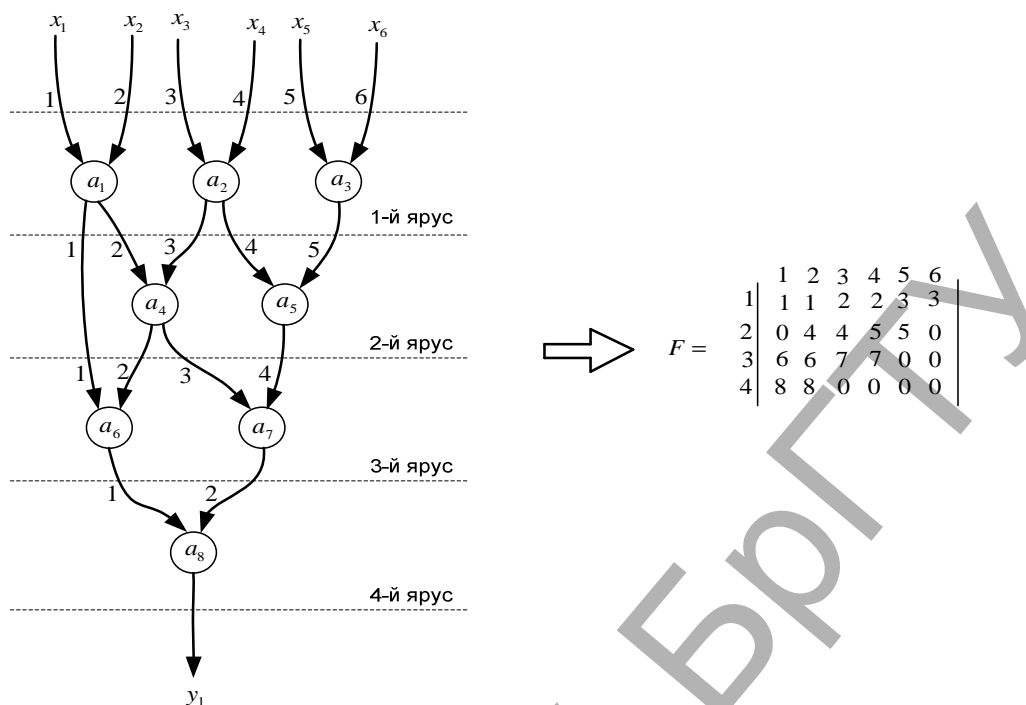


Рис. 4. Матричное преобразование ПГА с помощью структурной матрицы

Общее количество элементов матрицы инцидентности (S_{MI}) равно произведению количества вершин (N) на количество дуг (M) поточного графа алгоритма.

$$S_{MI} = N \times M. \quad (3)$$

Размер образованной матрицы на рис. 3 составляет 136 элементов.

Недостатком данного метода является то, что он требует большого объема для хранения матрицы (размер пропорционален количеству вершин и дуг ПГА) и поэтому неудобен для хранения. Кроме того, матрица инцидентности довольно редко используется, поскольку для графов с большим количеством ребер она имеет очень большое количество столбцов.

Представление ПГА структурной матрицей [11, 12] заключается в нумерации вершин и дуг ПГА по правилу: 1) в самом широком ярусе присвоить каждой дуге ПГА число $j, j = 1, n$, где n – количество входных дуг ПГА; 2) выполнить нумерацию дуг всех остальных ярусов по правилу: номера входных и выходных дуг для любой вершины ПГА одинаковы и совпадают; и в записи информации о ПГА структурной матрицей, количество столбцов которой равно количеству дуг в самом широком ярусе ПГА, а количество строк – общему количеству ярусов ПГА. Элементами такой матрицы являются номера ФО, которые записывают в элемент ij S_{CM} , где i – номер яруса, а j – номер дуги, которая поступает в данный ФО.

На рис. 4 показан переход от ПГА к структурной матрице (F).

Число элементов структурной матрицы (S_{CM}) равно произведению количества ярусов (l) на количество дуг самого широкого яруса (n) поточного графа алгоритма.

$$S_{CM} = l \times n. \quad (4)$$

Размер образованной матрицы на рис. 4 составляет 24 элемента.

Использование структурной матрицы позволяет сохранить распределение ФО по ярусам ПГА и связи между ними в удобном для обработки компьютером матричном виде, что дает возможность автоматизировать процесс сохранения, исследования и анализа ПГА.

Результаты исследования. Результаты исследования матричных способов преобразования поточных графов алгоритма матри-

цами инцидентности, матрицами смежности, структурной матрицей и матрицей связности портов и дуг для алгоритмов сортировки по методам Бетчера, модифицированным методом "пузырька", методом четно-нечетной перестановки и алгоритмов матричного умножения двоичных чисел с горизонтальным и диагональным распространением переноса приведены в таблице 1.

В таблице 1 в колонках представлены значения необходимых параметров для поточных графов выбранных алгоритмов (количество входных данных, количество вершин, количество дуг, количество ярусов, количество пар входных и выходных портов и количество дуг, соединяющих пары входных и выходных портов) и сделано расчет общего количества элементов исследуемых матриц по соответствующим формулам.

Для алгоритма сортировки модифицированным методом "пузырька" для n входных значений (данных) количество вершин (операций) составляет $n(n-1)/2$, количество дуг – $2N+n$, где N – количество вершин (операций), количество ярусов – $2n-3$, количество дуг соединяющих пары входных и выходных портов – $2N-n$.

Для алгоритма сортировки методом четно-нечетной перестановки для n входных значений (данных) количество вершин (операций) составляет $n(n-1)/2$, количество дуг – $2N+n$, где N – количество вершин (операций), количество ярусов – n , количество дуг, соединяющих пары входных и выходных портов – $2N-n$.

Для алгоритма сортировки методом Бетчера для n входных значений (данных) количество вершин (операций) составляет $0,48n \ln^2 n$, количество дуг – $2N+n$, где N – количество вершин (операций), количество ярусов – $\left(\frac{1}{2}[\log_2 n][\log_2 n + 1]\right)$,

количество дуг соединяющих пары входных и выходных портов – $2N-n$.

Для алгоритма матричного умножения двоичных чисел с горизонтальным распространением переноса для n входных значений (данных) количество вершин (операций) составляет n^2-n , количество дуг – $2N+(n^2-1)$, где N – количество вершин (операций), количество ярусов – $3n-3$, количество дуг, соединяющих пары входных и выходных портов – $2N-(2n-1)$.

Таблица 1. Размеры матриц исследуемых поточных графов алгоритмов

Параметры ПГА						Общее количество элементов матрицы			
К-во вх. данных, n	К-во вершин, N	К-во дуг, M	К-во ярус., l	К-во пар вх/вых. портов, p	К-во дуг, что соедин. пары вх/вых. портов, d	Инцидентности, $N \times M$	Смежности, $N \times N$	Структурная матрица, $l \times n$	Матрица связности портов и дуг, $p \times d$
сортировка модифицированным методом "пузырька"									
4	6	16	5	3	8	96	36	20	24
8	28	64	13	3	48	1792	784	104	144
16	120	256	29	3	224	30720	14400	464	672
32	496	1024	61	3	960	507904	246016	1952	2880
64	2016	4096	125	3	3968	8257536	4064256	8000	11904
128	8128	16384	253	3	16128	133169152	66064384	32384	48384
256	32640	65536	509	3	65024	2139095040	1065369600	130304	195072
512	130816	262144	1021	3	261120	34292629504	17112825856	522752	783360
1024	523776	1048576	2045	3	1046528	549218942976	274341298176	2094080	3139584
сортировка методом четно-нечетной перестановки									
4	6	16	4	4	8	96	36	16	32
8	28	64	8	4	48	1792	784	64	192
16	120	256	16	4	224	30720	14400	256	896
32	496	1024	32	4	960	507904	246016	1024	3840
64	2016	4096	64	4	3968	8257536	4064256	4096	15872
128	8128	16384	128	4	16128	133169152	66064384	16384	64512
256	32640	65536	256	4	65024	2139095040	1065369600	65536	260096
512	130816	262144	512	4	261120	34292629504	17112825856	262144	1044480
1024	523776	1048576	1024	4	1046528	549218942976	274341298176	1048576	4186112
сортировка методом Бетчера									
4	9	22	3	4	14	198	81	12	56
8	19	46	6	4	30	874	361	48	120
16	63	142	10	4	110	8946	3969	160	440
32	195	422	15	4	358	82290	38025	480	1432
64	543	1150	21	4	1022	624450	294849	1344	4088
128	1467	3062	28	4	2806	4491954	2152089	3584	11224
256	3839	7934	36	4	7422	30458626	14737921	9216	29688
512	9634	19780	45	4	19268	190560520	92813956	23040	77072
1024	24063	49150	55	4	47102	1182696450	579027969	56320	188408
матричное умножения двоичных чисел с горизонтальным распространением переноса									
4	12	31	9	3	17	468	144	36	51
8	56	175	21	3	97	9800	3136	168	291
16	240	735	45	3	449	176400	57600	720	1347
32	992	3007	93	3	1921	2983936	984064	2976	5763
64	4032	12159	189	3	7937	49025088	16257024	12096	23811
128	16256	48895	381	3	32257	794837120	264257536	48768	96771
256	65280	196095	765	3	130049	12801081600	4261478400	195840	390147
512	261632	785407	1533	3	522241	205487604224	68451303424	784896	1566723
1024	1047552	3143679	3069	3	2094081	3293167223808	1097365192704	3142656	6282243
матричное умножения двоичных чисел с диагональным распространением переноса									
4	12	31	8	3	17	468	144	32	51
8	56	175	16	3	97	9800	3136	128	291
16	240	735	32	3	449	176400	57600	512	1347
32	992	3007	64	3	1921	2983936	984064	2048	5763
64	4032	12159	128	3	7937	49025088	16257024	8192	23811
128	16256	48895	256	3	32257	794837120	264257536	32768	96771
256	65280	196095	512	3	130049	12801081600	4261478400	131072	390147
512	261632	785407	1024	3	522241	205487604224	68451303424	524288	1566723
1024	1047552	3143679	2048	3	2094081	3293167223808	1097365192704	2097152	6282243

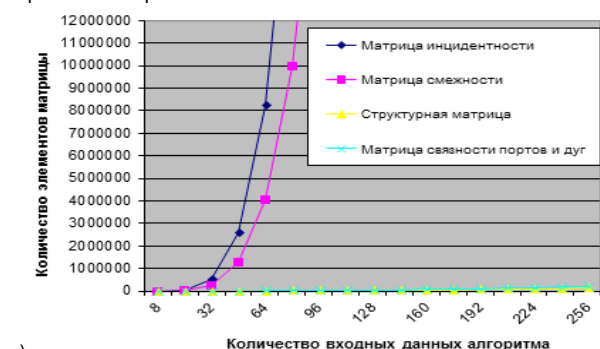
Для алгоритма матричного умножения двоичных чисел с диагональным распространением переноса для n входных значений (данных) количество вершин (операций) составляет $n^2 - n$, количество дуг – $2N + (n^2 - 1)$, где N – количество вершин (опера-

ций), количество ярусов – $2n$, количество дуг, соединяющих пары входных и выходных портов – $2N - (2n - 1)$.

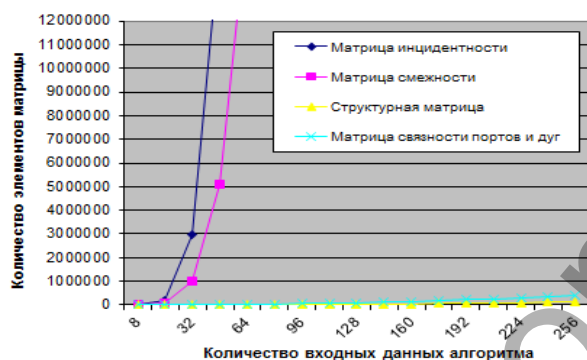
В таблице 1 видим, что с увеличением количества входных данных соответственно возрастает и общее количество элементов мат-

риц, которые являются наибольшей для матриц инцидентности и матриц смежности и значительно меньше для структурной матрицы и матрицы связности портов и дуг, а также наблюдается постоянное значение количества пар входных и выходных портов матрицы связности портов и дуг для каждого алгоритма, которое не зависит от количества входных данных.

На рис. 5 представлены графики зависимостей количества элементов исследуемых матриц от количества входных данных для алгоритмов сортировки модифицированным методом "пузырька" и матричного умножения двоичных чисел с диагональным распространением переноса.



а)



б)

Рис. 5. Сравнение размеров матриц ПГА: а) для алгоритма сортировки модифицированным методом "пузырька"; б) для алгоритма матричного умножения двоичных чисел с диагональным распространением переноса

На графике можно видеть, что при увеличении количества входных данных ПГА количество элементов матриц инцидентности и смежности экспоненциально возрастает, а количество элементов структурной матрицы и матрицы связности портов и дуг возрастает линейно.

Необходимо отметить, что для сохранения в памяти компьютера соответствующей матрицы необходимо учитывать размер ячейки памяти для записи одного элемента матрицы. Для матрицы смежности и матрицы связности портов и дуг размер ячейки составляет один бит, поскольку нужно выполнить запись 0 или 1, для матрицы инцидентности необходимо два бита, поскольку необходимо записать множество значений $\{0, 1, -1\}$, и для структурной матрицы размер ячейки равен количеству битов, необходимых для записи значения, которое равно сумме вершин ПГА.

Заключение. Исследование матричных способов задания ПГА дает возможность оценить их преимущества и недостатки по таким основным

параметрам, как объем памяти необходим для сохранения соответствующей матрицы в памяти компьютера, размер ячейки памяти для записи одного элемента соответствующей матрицы и возможности автоматизировать процесс сохранения, исследования и анализа ПГА.

Матрицы инцидентности и смежности успешно применять для описания ПГА при небольших количествах вершин и дуг. Структурная матрица за счет записи упорядоченного распределения вершин по ярусам и связей между ними хранит информацию о ПГА и уменьшает объем памяти, необходимый для сохранения ПГА, поскольку размер СМ зависит от количества ярусов, по которым эти вершины закреплены, и от количества дуг самого широкого яруса. Матрица связности портов и дуг за счет постоянного значения количества пар входных и выходных портов которые не зависят от количества входных данных, занимает значительно меньший объем памяти по отношению к матрицам инцидентности и смежности и сравнительно со структурной матрицей требует меньшего размера ячейки памяти для записи одного элемента за счет бинарного множества своих значений.

Результаты исследований показали, что с увеличением количества входных данных алгоритма количество ячеек памяти для сохранения поточных графов исследуемых алгоритмов с помощью матриц инцидентности и смежности экспоненциально увеличиваются, в то время как количество ячеек памяти для сохранения структурной матрицы и матрицы связности портов и дуг увеличивается линейно.

СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Ахо, А. Построение и анализ вычислительных алгоритмов / А. Ахо, Дж. Хопкрофт, Жд. Ульман. – М.: Мир, 1979. – 536 с.
2. Дунець, Р.Б. Аналіз та синтез топологій комп'ютерних видавничо-поліграфічних систем: монографія – Львів: НВФ "Українські технології", 2003. – 145 с.
3. Глушков, В.М. Синтез цифровых автоматов / В.М. Глушков – М.: Физматгиз, 1962 – 476 с.
4. Мельник, А.О. Спеціалізовані комп'ютерні системи реального часу – Львів: ДУ "ЛП", 2002. – 60 с.
5. Вишенчук, И.М. Алгоритмические устройства и супер ЭВМ / И.М. Вишенчук, Н.В. Черкасский – К.: Техника, 1991. – 197 с.
6. Мельник, А.О. Архітектура комп'ютера. Наукове видання. – Луцьк: Волинська обласна друкарня, 2008. – 470 с.
7. Ерметов, Ю.О. Проектування обчислювальних структур на основі просторово-часових графів. // Вісн. Хмельницького національного університету. – Хмельницький, 2006. – № 4. – С. 172–177.
8. Дунець, Р.Б. Застосування просторово-часових графів для синтезу спеціалізованих перемножувачів / Р.Б. Дунець, В.М. Грига // Науково-технічний журнал "Радіоелектронні і комп'ютерні системи" – Харків: Національний аерокосмічний університет ім. М.Є. Жуковського "Харківський авіаційний інститут", 2008. – № 7(34) – С. 113–118.
9. Кормен, Т.Х. Алгоритмы: построение и анализ / Т.Х. Кормен, Ч.И. Лейзерсон, Р.Л. Ривест, К. Штайн – М.: Вильямс, 2005. – 1296 с.
10. Кун, С. Матричные процессоры на СБИС. – М.: Мир, 1991. – 754 с.
11. Мельник, А.О. Подання потокового графа алгоритму структурною матрицею / А.О. Мельник, І.Д. Яковлева // Науковий журнал "Технічні науки" – Хмельницький: Хмельницький національний університет, 2008. – № 4 – С. 124–129.
12. Мельник, А.О., Яковлева І.Д. Порівняльний аналіз способів матричного подання графа алгоритму / А.О. Мельник, І.Д. Яковлева // Вісник "Комп'ютерні системи та мережі". – Львів: Національний університет "Львівська політехніка", 2009. – № 658. – С. 37–45.

Материал поступил в редакцию 10.10.13

DUNEETS R.B., GRIEGA V.M. Research of matrix methods of representation of line counts of algorithms

The matrix conversion method flow graph structure algorithm using a matrix of connectivity ports and arcs. This approach allows using matrix transformations move from flow graph structure to the basic structures of space-time graph algorithms that are used to design multitact specialized device. A comparative analysis of the known matrix methods default flow graph of the algorithm with the proposed matrix method in this paper. The result shows that the matrix of connectivity ports and arcs takes much less memory than incidence matrix and adjacency and is commensurate with the structural matrix.