
УДК 658.012

Л.И. Тимченко, д-р техн. наук, **Н.И. Кокряцкая**, канд. техн. наук
Государственный экономико-технологический университет транспорта
(Украина, 03049, Киев, ул. Лукашевича, 19,
тел.: +380673550976, +380432531123,
e-mail: timchen@list.ru, kokriatskaia@rambler.ru),

М.П. Поддубецкая
Винницкий национальный технический университет
(Украина, 21021, Винница, ул. Хмельницкое шоссе, 95,
тел. +380969644470, e-mail: marynar777@gmail.com)

Метод моделирования параллельно-иерархической сети для обработки данных на основе построения функциональных рядов

Представлен метод моделирования параллельно-иерархической сети на основе функциональных рядов. Разработано программное обеспечение, позволяющее моделировать G -преобразования на каждом уровне сети в соответствии с ранее выбранными элементами, а также программное обеспечение, моделирующее базовую сеть произвольной размерности посредством формирования его функциональных рядов. Полученные результаты можно использовать для решения задач обработки и систематизации больших объемов данных, в том числе графических.

Подано метод моделювання паралельно-ієрархічної мережі на основі функціональних рядів. Розроблено програмне забезпечення, яке дозволяє моделювати G -перетворення на кожному рівні мережі відповідно до раніше обраних елементів, а також програмне забезпечення, яке моделює базисну мережу довільної розмірності за допомогою формування її функціональних рядів. Отримані результати можна використовувати для вирішення задач обробки і систематизації великих об'ємів даних, в тому числі графічних.

Ключевые слова: параллельно-иерархическая сеть, функциональный ряд, базовая сеть, хвостовой элемент.

Параллельно-иерархическая (ПИ) сеть представляет собой технологию машинного самообучения, которая является некоторым подобием структурных и алгоритмических свойств коры головного мозга человека. Формирование многоэтапной ПИ сети предполагает процесс последовательного преобразования пространственных областей коррелированных и образования декорреляционных во времени элементов нейронной сети при переходе ее из одного устойчивого состояния в другое.

© Л.И. Тимченко, Н.И. Кокряцкая, М.П. Поддубецкая, 2015

В машинной графике и обработке изображений широко используются иерархические структуры данных [1—3] для моделирования и описания изображений. Необходимость в многоуровневых представлениях обусловлена появлением задач, для решения которых требуется упрощение абстрактных описаний и дискретное представление сложных структур данных (с сохранением важных топологических свойств). Именно поэтому при проведении операций над объектами или областями изображений возникает необходимость в разработке математического аппарата, который позволил бы выполнять действия над частями изображений как над элементарными объектами.

Иерархическое представление позволяет избежать необходимости отдельной обработки каждого элемента входных данных, так как ПИ структуры дают возможность представить в качестве элементарных такие части входных данных, которые одновременно имеют размер, вес и положение во времени и пространстве, т.е. являются блоками различных уровней сети. Именно поэтому остается актуальной задача разработки ПИ сетей для описания данных, например, в задачах кластеризации и сегментации статических изображений.

Основными источниками для разработки методики и формулировки принципов обработки информации в ПИ сети являются результаты нейрофизиологических исследований, открывающие новые научные взгляды на распространение сигналов в мозге [4]. Сигналы поступают к центру петлеобразно с задержкой, связывая различные участки мозга в единую сеть. Как отмечено в работе [5], «вычисления» изображения в мозгу происходят по параллельным восходящим потокам данных и переходят из одного нейронного участка в другой по иерархическому принципу.

Принципы параллельности, иерархичности и задержки во времени прохождения сигналов — методологическая основа организации ПИ сети, прототипом которой можно считать широко известную нейронную сеть Time-Delay Neural Network [6]. Обработка данных выполняется на основе оригинального метода [7], с использованием разработанной ПИ сети, в которой сначала выполняется параллельная обработка информации в горизонтальных каналах, а затем в вертикальных. Далее процесс обработки с задержкой во времени переходит на следующий уровень, где происходит такое же чередование параллельной обработки в горизонтальных и вертикальных каналах.

Рассмотрим метод моделирования ПИ сети на основе анализа функциональных рядов с использованием разработанного программного обеспечения для синтеза информационных структур и вычислительных устройств на основе ПИ сетей.

Метод моделирования ПИ сети на основе функциональных рядов.

Параллельно-иерархическая сеть состоит из ряда уровней, а каждый уровень — из параллельных ветвей. Начиная со второго уровня, каждая последующая ветвь сдвинута во времени относительно сложившейся предыдущей ветви. Входной образ подается на элементы параллельных ветвей первого уровня, а элементы ветвей второго и последующих уровней формируются последовательно во времени в соответствии с элементами ветвей первого и предыдущих уровней. Длина алгоритма ПИ сети и соответственно число элементов в ветвях, число ветвей и уровней — величины не постоянные, а зависят от характера распределения входных данных.

Основным назначением ПИ сети является решение задачи быстрого преобразования, классификации и прогнозирования, ориентированной на обработку видеоданных в реальном времени. Анализ образа заключается в последовательном преобразовании совпадающих и выявлении (фильтрации) несовпадающих во времени составляющих образа при переходе элементов нейронной сети из текущих энергетических состояний с одними пространственными координатами в состояния с меньшей энергией и другими пространственными координатами.

Условием перехода составляющих образа на следующий уровень ПИ сети является наличие динамики взаимного совпадения промежуточных результатов обработки во времени с одноименными каналами нижнего уровня. Результат анализа образа формируется из изолированных в пространственно-временной области составляющих образа [7].

Параллельно-иерархические сети позволяют анализировать образы и данные, систематизировать их, наблюдать за взаимным совпадением результатов во времени. В нейронных связях чередование ветвей сети (горизонтальный путь, вертикальный путь, горизонтальный путь и т.д.) создает временной сдвиг, вследствие которого и формируется иерархическая структура ПИ сети [8—10].

Составим функциональное описание базовой сети. Пусть имеется информационный поток, заданный в виде множеств, где n — размерность i -го множества. В общем случае базовую сеть можно описать с помощью шести разновидностей функциональных рядов:

$$C_1(i, j) = \bigcup_{i=1}^n \bigcup_{j=1}^n \left(M \left[\frac{n-(i-1)}{j} \right] \right). \quad (1)$$

$$S_1(i, j) = \bigcup_{i=n+1}^{2n-1} \bigcup_{j=2}^i \left(M \left[\frac{i}{j} \right] \right), \quad S_1(i, j) = \bigcup_{i=n+1}^{2n-1} M(2n-i), \quad j = 1, \quad (2)$$

$$S_2(i, j) = \bigcup_{i=2n}^{3(n-1)} \bigcup_{j=4}^i \left(M \left[\begin{matrix} i \\ j \end{matrix} \right] \right), \quad S_2(i, j) = \bigcup_{i=2n}^{3(n-1)} M(3n-2-i), \quad j=3, \quad (3)$$

$$S_3(i, j) = \bigcup_{j=5}^{3n-2} \left(M \left[\begin{matrix} 3n-2 \\ j \end{matrix} \right] \right), \quad i=3n-2, \quad (4)$$

$$S_3(i, j) = M(1), \quad i=3n-2, \quad j=3, \quad S_3(i, j) = M(1), \quad i=3n-2, \quad j=4,$$

$$S_4(i, j) = \bigcup_{i=3n-1}^{3(n-2)+(n-3)^2} \bigcup_{j=i+(7-3n)}^i \left(M \left[\begin{matrix} i \\ j \end{matrix} \right] \right), \quad S_4(i, j) = M(1),$$

$$i = \overline{(3n-1), (3n-2)+(n-3)^2}, \quad j = \overline{5, 4+(n-3)^2}, \quad (5)$$

$$S_5(i, j) = \bigcup_{i=(3n-1)+(n-3)^2}^{n^2} \bigcup_{j=2i-(n-3)^2+9-6n}^i \left(M \left[\begin{matrix} i \\ j \end{matrix} \right] \right), \quad S_5(i, j) = M(1),$$

$$i = \overline{(3n-1)+(n-3)^2, n^2}, \quad j = \overline{(n-3)^2+6, (n-3)^2+2(3n-5)}, \quad (6)$$

$$C_2(i, j) = S_1(i, j) \cup S_2(i, j) \cup S_3(i, j) \cup S_4(i, j) \cup S_5(i, j),$$

$$C(i, j) = C_1(i, j) \cup C_2(i, j), \quad (7)$$

где j — уровень иерархии множества $M_i(n)$ в базисной сети; $\lceil \cdot \rceil$ — выделение целой части и округление до большего значения.

Число рядов вида (1) определяется размером n . Более просто можно описать базисную сеть, если ряды (1)—(6) давать парами. Например, ряд $S_1(i, j)$ описывается первым выражением формулы (2) при $j=2$, i и вторым выражением при $j=1$. Таким образом, первые элементы рядов $S_1(i, j)$ отражают состояния базовой сети для первого уровня ($j=1$). Другие элементы формируются из первого выражения (2) и отражают состояния базовой сети на следующих уровнях. Число рядов вида (2) составляет $n-1$, число рядов вида (3) — $n-2$. Очевидно, что число рядов вида (4) ограничивается одной последовательностью. Число рядов вида (5) не является постоянной величиной для различных значений n . Если $n < 4$, то данный ряд вообще отсутствует. Если $n \geq 5$, то число таких рядов составляет $n-1$. Число рядов вида (6) также изменяется в зависимости от значения n : например, для $n=3$ число рядов равно двум, для $n=4$ — шести, а для $n=5$ — восьми.

Ряды (1)—(6) в общем виде описывают структуру ПИ сети. Функциональный ряд $C_1(i, j)$ образует основную базовую сеть, а ряды $C_2(i, j)$ — хвостовую. Их совокупность образует полную базовую сеть. Таким обра-

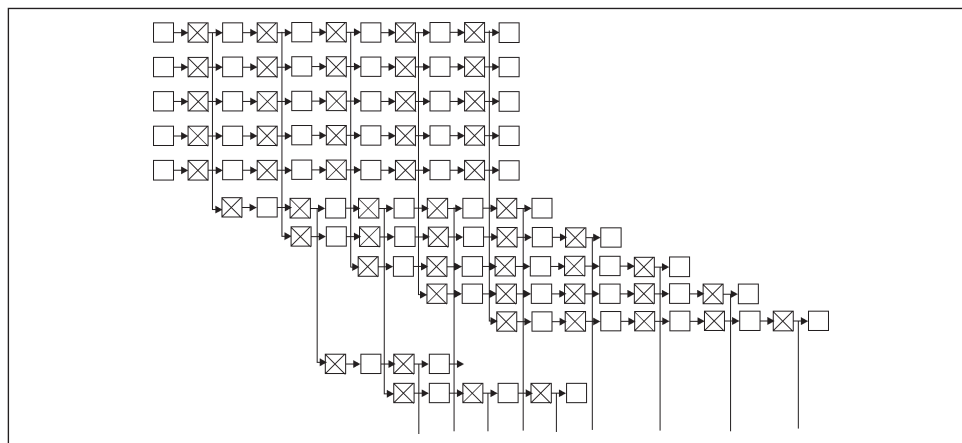


Рис. 1. Фрагмент структурной схемы сети размерностью 5×5 : \boxtimes — промежуточный элемент; \square — множество элементов

зом, базовая ПИ сеть состоит из основной и хвостовой сетей, которые являются основными составляющими взаимодействия параллельных потоков данных.

Согласно (1)—(6) ПИ преобразование полной базовой сети заключается в следующем. На первом уровне параллельными средствами в независимых ветвях (каналах) выполняется исходное (заданное для конкретной задачи) преобразование, в результате которого формируются информационные потоки данных для преобразований на следующих уровнях. На каждом следующем уровне происходит формирование хвостового элемента сети — последнего элемента функциональных рядов (1)—(6) для каждого значения i и нового информационного потока, являющегося исходным для преобразования на следующем уровне сети.

Пример работы программы ПИ сети на основе функциональных рядов. Разработанное программное обеспечение позволяет промоделировать базовую сеть произвольной размерности посредством формирования функциональных рядов. На первом уровне выполняется исходное преобразование, создаются информационные потоки, преобразуемые на основе (1)—(6) и реализуемые на следующих уровнях. На каждом уровне формируются хвостовые элементы, которые выводятся на экран.

Рассмотрим работу программы для размерности сети $n = 5$. На рис. 1 представлена структурная схема сети, в которой первый элемент каждого уровня, начиная со второго, является хвостовым элементом. Формирование элементов ряда S_1 программным приложением по формуле (1) показано в таблице. Сформированные значения хранятся в памяти и происхо-

$i=6$	$i=7$	$i=8$	$i=9$	j
3	4	4	5	2
2	3	3	3	3
2	2	2	3	4
2	2	2	2	5
1	2	2	2	6
0	1	2	2	7
0	0	1	2	8
0	0	0	1	9

$C_1(i, j)$	}	1
		2 1
		3 2 1
		4 2 2 1
		5 3 2 2 1
$S_1(i, j)$	}	4 3 2 2 2 1
		3 4 3 2 2 2 1
		2 4 3 2 2 2 2 1
		1 5 3 3 2 2 2 2 1
		3 3 2 2 2 2 2 1
$S_2(i, j)$	}	2 3 3 2 2 2 2 2 1
		1 3 3 2 2 2 2 2 2 1
$S_3(i, j)$	{	1 1 3 3 2 2 2 2 2 2 1
$S_4(i, j)$	}	1 3 2 2 2 2 2 2 2 2 1
		1 3 2 2 2 2 2 2 2 2 1
		1 2 2 2 2 2 2 2 2 2 1
		1 2 2 2 2 2 2 2 2 2 1
$S_5(i, j)$	}	1 2 2 2 2 2 2 2 2 1
		1 2 2 2 2 2 2 2 1
		1 2 2 2 2 2 2 1
		1 2 2 2 2 2 1
		1 2 2 2 1
		1 2 1
		1 1

Рис. 2. Формирование рядов базовой сети размерностью 5×5

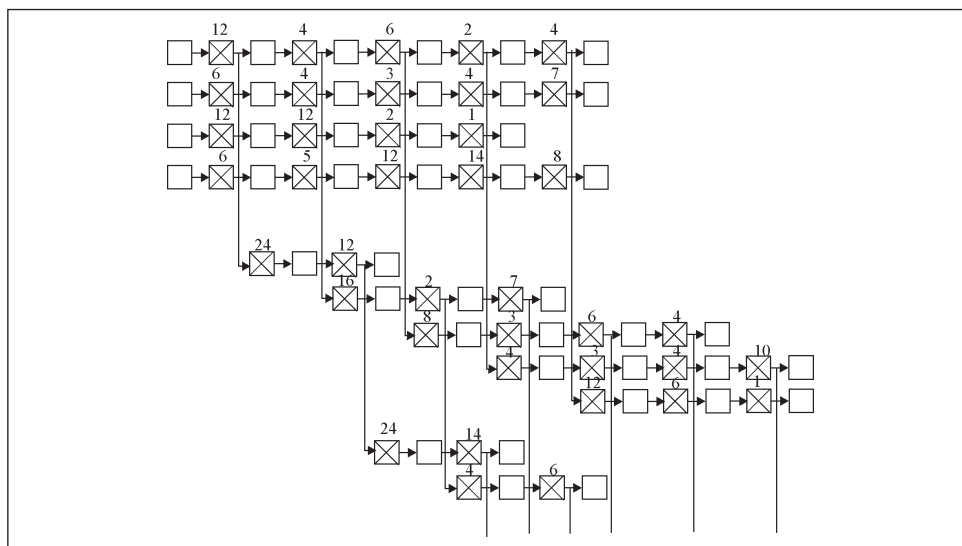


Рис. 3. Фрагмент структурной схемы работы программы, моделирующей процесс сетевого ПИ преобразования

дит дальнейшая обработка функциональных рядов. При $j = 1$ программа подсчитывает значения ряда S_1 по формуле

$$S_1(i, j) = \bigcup_{i=n+1}^{2n-1} M(2n-i).$$

Результатом работы программы является вывод на экран полной базовой сети, состоящей из совокупности рядов $C_1(i, j)$ и $C_2(i, j)$ (рис. 2). В каждом такте на любом уровне происходит выбор элементов из множеств по определенному критерию и реализуются так называемые G -преобразования [8] множеств в соответствии с ранее выбранными элементами, что свидетельствует о свойстве синхронности данной сети. Каждый промежуточный элемент (см. рис. 1) не входит ни в одно множество, потому что в данном такте он для своего уровня выбран как единственный не участвующий в дальнейшей обработке массивов и является его выходным результатом. Именно поэтому такие элементы называются хвостовыми или элементами, формирующими результат.

Разработано программное обеспечение, которое позволяет смоделировать работу сети и выполнить преобразования множеств в соответствии с ранее выбранными элементами, а также формирование хвостовых элементов, не участвующих в дальнейшем преобразовании. Описанный процесс сетевого ПИ преобразования схематически представлен на рис. 3.

Процесс формирования элементов сети, где C_n — исходное множество элементов, m — выбранный наименьший элемент данной итерации, представим в следующем виде:

C_1	m	C_1	m	C_1	m	C_1	m	C_1	m
2		0		0		0		0	
3	2	1	1	0	2	0	1	0	4
5	↓	3	↓	2	↓	0	↓	0	↓
6	12	4	4	3	6	1	2	0	4
10		8		7		5		4	
2		0		0		0		0	

Элементы 12, 4, 6, 2, 4 формируют первую ветвь первого уровня ПИ сети и базовое множество для преобразования на ее втором уровне (см. рис. 3). Аналогично происходит формирование других ветвей первого уровня ПИ сети:

C_2	m	C_2	m	C_2	m	C_2	m	C_2	m
1		0		0		0		0	
5	1	4	1	3	1	2	2	0	7
2	↓	1	↓	0	↓	0	↓	0	↓
12	6	11	4	10	3	9	4	7	7
3		2		1		0		0	
1		0		0		0		0	

C_3	m	C_3	m	C_3	m	C_3	m
2		0		0		0	
5	2	3	3	0	1	0	1
6	↓	4	↓	1	↓	0	↓
7	12	5	12	2	2	1	1
2		0		0		0	
5		0		0		0	

C_4	m	C_4	m	C_4	m	C_4	m	C_4	m
5		4		3		0		0	
12	1	11	1	10	3	7	7	0	8
20	↓	19	↓	18	↓	15	↓	8	↓
2	6	1	5	0	12	0	14	0	8
5		4		3		0		0	
1		0		0		0		0	

Формирование элементов второго уровня, где A_n — исходное множество элементов, происходит так:

A_1	m	A_1	m				
12		6					
6	6	0	6				
12	↓	6	↓				
6	24	0	12				
(первый хвостовой элемент)							
A_2	m	A_2	m	A_2	m		
4		0		0			
4	4	0	1	0	7		
12	↓	8	↓	7	↓		
5	16	1	12	0	7		
A_3	m	A_3	m	A_3	m	A_3	m
6		4		3		0	
3	2	1	1	0	3	0	6
2	↓	0	↓	0	↓	0	↓
12	8	10	3	9	6	6	6
A_4	m	A_4	m	A_4	m	A_4	m
2		1		0		0	
4	1	3	1	2	2	0	10
1	↓	0	↓	0	↓	0	↓
14	4	13	3	12	4	10	10
A_4	m	A_4	m	A_4	m		
4		0		0			
7	4	3	3	0	1		
8	↓	4	↓	1	↓		
	12		6		1		

Формирование элементов третьего уровня, где B_n — исходное множество элементов, происходит аналогично:

- 3) вывод хвостовых элементов (результата), построение структуры сети;
 4) сравнение соответствующих данных элементов эталонного и текущего изображений с использованием элементов функциональных рядов.

Разработан программный продукт, осуществляющий выбор элементов из множеств по определенному критерию, например по минимальному значению, или так называемые G -преобразования множеств в соответствии с ранее выбранными элементами. Математическая модель такого преобразования заключается в параллельной обработке каждого массива значений, который обрабатывается последовательно на каждом уровне, а также последовательно обрабатываются массивы на разных уровнях, т.е. формирование следующего массива для G -преобразования возможно только в том случае, если преобразования на предыдущих уровнях уже состоялись.

Рассмотрим подробнее процесс моделирования G -преобразования на каждом уровне ПИ сети. Пусть имеются ряды $A_1 = (6, 6, 3)$, $A_2 = (6, 4, 4)$, $A_3 = (0, 1, 16)$. На первом шаге алгоритма выбирается наименьшее значение из множества. Для A_1 это значение равно трем. От каждого элемента множества вычитается выбранное наименьшее значение и результат записывается в новое множество A_{12} . В то же время выбранное на предыдущей итерации наименьшее значение умножается на число ненулевых значений в множестве и запоминается как элемент нового множества A_1'' . Последовательность действий выполняется до тех пор, пока во множестве A_{1n} все значения не станут нулевыми. Записанные элементы множества A_1'' — это хвостовые элементы, которые переходят на следующий уровень сети для выполнения нового G -преобразования.

Пример G -преобразования:

A_1	m	A_{12}	m
6		3	
6	3	3	3
3	↓	0	↓
	9		6
A_2	m	A_{22}	m
6		2	
4	4	0	2
4	↓	0	↓
	12		2
A_3	m	A_{32}	m
0		0	15
1	1	15	↓
16	↓		15
	2		

Как показали экспериментальные исследования, программное средство реализует следующие функции:

- 1) выполнение процесса преобразования множеств в соответствии с ранее выбранными элементами, а также формирование хвостовых элементов, не участвующих в дальнейшем преобразовании;
- 2) расчет числа итераций при работе в среде моделирования;
- 3) просмотр каждой итерации процесса кодирования информационных массивов методом прямого ПИ преобразования;
- 4) вывод хвостовых элементов (результата).

Для работы с программной средой необходимо:

- запустить выполнение файла `Paralel_G-convertation.exe`;
- загрузить информационный массив из файла (в формате `txt`) или заполнить поля ввода (рандомизированное заполнение полей для ввода данных обычным натуральным рядом, реализована возможность задания пользователем размерности входного информационного массива данных);
- осуществить просмотр (при необходимости) каждой итерации прямого ПИ преобразования для анализа работы программы.

Выводы

Разработанный метод ПИ сети на основе функциональных рядов является основой организации сравнения изображений и реализуется посредством сравнения соответствующих элементов функциональных рядов эталонного и текущего изображений. Используя функциональное описание ПИ сети посредством анализа промежуточных и хвостовых ее элементов, представленных элементами функциональных рядов, можно достаточно просто (без выполнения трудоемких вычислительных операций) проводить сравнение входных данных, в частности изображений. Полученные результаты могут быть использованы для решения широкого круга задач, требующих выполнения сложных операций, например сравнения или поиска частей цифровых изображений в системах связи и передачи информации, операций с большими объемами данных [12].

СПИСОК ЛИТЕРАТУРЫ

1. *Ruei-Yu Wu, Gen-Huey Chen, Jung-Sheng Fu et al.* Finding cycles in hierarchical hypercube networks // *J. Information Proc. Letters*. — 2008. — Vol. 109, Iss. 2. — P. 112—115.
2. *Sven Behnke* Hierarchical Neural Networks for Image Interpretation. — Berlin—Heidelberg — N Y: Springer-Verlag, 2003.
3. *Srivastava L., Singh S.N., Sharma J.* Parallel self-organising hierarchical neural network-based fast voltage estimation//*IEEE Proc. — Generation Transmission and Distribution*. — 1998. — Vol. 145, Iss. 1. — P. 98—104.
4. *Middleton V.T.C., Hawkins R.* Sustainable Tourism: A Marketing Perspective. — Oxford: Butterworth — Heinemann, 1998.

5. Thompson R.H., Swanson L.W. Hypothesis-driven structural connectivity analysis supports network over hierarchical model of brain architecture// Proc. of the National Academy of Sciences USA. — 107 (34). — P. 15235—15239. Published online 2010 August 9. doi: 10.1073/pnas.1009112107.
6. Lafer-Sousa Rosa, Conway Bevil R. Parallel, multi-stage processing of colors, faces and shapes in macaque inferior temporal cortex // Nature Neuroscience. — 2013. — № 16. — P. 1870—1878.
7. Kaiser M. Time-Delay Neural Networks for Control // 4th International Symposium on Robot Control (SYROCO'94), Capri, Italy, 1994.
8. Тимченко Л.И., Мельников В.В., Кокряцкая Н.И. и др. Метод организации параллельно-иерархической сети для распознавания образов// Кибернетика и системный анализ. — 2011. — № 1. — С. 152—163.
9. Тимченко Л.И., Мельников В.В., Кокряцкая Н.И. и др. Применение параллельно-иерархического метода распознавания изображений пятен лазерных пучков // Материалы Междунар. научно-техн. конф. «Многопроцессорные вычислительные и управляющие системы». Таганрог, 28 сентября — 3 октября 2009 г. — Таганрог: ЮФУ, 2009. — Т. 2. — С. 147—149.
10. Тимченко Л.И. Конвергентні та дивергентні процеси в реальних та штучних нейронних мережах // Вісник ВПІ. — 1997. — № 1. — С. 5—10.
11. Тимченко Л.И. Многоэтапная параллельно-иерархическая сеть как модель нейроподобной схемы вычислений//Кибернетика и системный анализ. — 2000. — № 2. — С. 114—134.
12. Timchenko L. I., Kokryatskaya N.I., Melnikov V.V., Kosenko G.L. Method of forecasting energy center positions of laser beam spot images using a parallel hierarchical network for optical communication systems //J. Opt. Eng. 52 (5), 055003 (May 09, 2013). — doi: 10.1117/1.OE.52.5.055003.

L.I. Timchenko, N.I. Kokryatskaya, M.P. Piddubetskaya

METHOD FOR MODELING PARALLEL- HIERARCHICAL NETWORK FOR PROCESSING DATA BASED ON THE CONSTRUCTION OF FUNCTIONAL SERIES

The method for modeling parallel-hierarchical network based on the functional series is presented. The software, which allows simulating G -transformation at every level of the network according to the previously chosen elements, has been developed. The software, which simulates the basic network of arbitrary dimension by creating its functional series, has been developed as well. The results may be used to solve problems of processing and organizing large bodies of data, including graphical ones.

Key words: parallel-hierarchical network, functional series, basic network, tail element.

REFERENCES

1. Ruei-Yu Wu, Gen-Huey Chen, Jung-Sheng Fu and Gerard J. Chang (2008), “Finding cycles in hierarchical hypercube networks”, *J. Information Processing Letters*, Vol. 109, Iss. 2, pp. 112-115.
2. Sven Behnke (2003), Hierarchical neural networks for image interpretation, Springer-Verlag, Berlin, Heidelberg, New York.
3. Srivastava, L., Singh, S.N. and Sharma, J. (1998), “Parallel self-organizing hierarchical neural network-based fast voltage estimation”, *IEEE Proceedings, Generation, Transmission and Distribution*, Vol. 145, Iss. 1, pp. 98-104.

4. Middleton, V.T.C. and Hawkins, R. (1998), Sustainable tourism: a marketing perspective, Butterworth-Heinemann, Oxford, UK.
5. Thompson, R.H. and Swanson, L.W. (2010), "Hypothesis-driven structural connectivity analysis supports network over hierarchical model of brain architecture", *Proceedings of the National Academy of Sciences USA*, Vol. 107, Iss. 34, pp. 15235-15239. Published online 2010 August 9, - doi: 10.1073/pnas.1009112107.
6. Lafer-Sousa, Rosa and Conway, Bevil R. (2013) "Parallel, multi-stage processing of colors, faces and shapes in macaque inferior temporal cortex", *Nature Neuroscience*, no. 16, pp. 1870-1878.
7. Kaiser, M. (1994), "Time-delay neural networks for control", *Proceedings of the 4th International Symposium on Robot Control (SYROCO'94)*, Capri, Italy.
8. Timchenko, L.I., Melnikov, V.V., Kokryatskaya, N.I. and Kutaev, Y.F. (2011), "The method of organization of parallel-hierarchical networks for pattern recognition", *Kibernetika i sistemny analiz*, no. 1, pp. 152-163.
9. Timchenko, L.I., Melnikov, V.V., Kokryatskaya, N.I. and et al. (2009), "The use of parallel-hierarchical method of image recognition spots of laser beams", *Materialy Mezhdunarodnoi nauchno-tekhnicheskoi konferentsii "Mnogoprotsessornye vychislitelnye i upravlyayushchie sistemy"* [Proceedings of International Scientific-Technical Conference «Multiprocessor Computing and Control Systems»], Taganrog, September 28-October 3, 2009, pp. 147-150.
10. Timchenko, L.I. (1997), "Processes in real and artificial neuron networks", *Visnyk VPI*, no.1, pp. 5-10.
11. Timchenko, L.I. (2000), "Multi-stage parallel-hierarchical network as a model of neural computing circuits", *Kibernetika i sistemny analiz*, no. 2, pp.114-134.
12. Timchenko, L.I., Kokryatskaya, N.I., Melnikov, V.V. and Kosenko, G.L. (2013), "Method of forecasting energy center positions of laser beam spot images using a parallel hierarchical network for optical communication systems", *J. Opt. Eng.*, Vol. 52, no. 5, 055003 (May 09, 2013), doi: 10.1117/1.OE.52.5.055003.

Поступила 05.11.15

ТИМЧЕНКО Леонид Иванович, д-р техн. наук, профессор, зав. кафедрой телекоммуникационных технологий и автоматики Государственного экономико-технологического университета транспорта. В 1979 г. окончил Винницкий политехнический ин-т. Область научных исследований — системы искусственного интеллекта.

КОКРЯЦКАЯ Наталья Ивановна, канд. техн. наук, доцент кафедры телекоммуникационных технологий и автоматики Государственного экономико-технологического университета транспорта. В 1973 г. окончила Винницкий педагогический ин-т. Область научных исследований — математическое моделирование и параллельная обработка информации.

ПОДДУБЕЦКАЯ Марина Петровна, магистрант кафедры программного обеспечения Винницкого национального технического университета. Область научных исследований — синтез и анализ изображений.