



DOI: <http://dx.doi.org/10.15688/jvolsu1.2016.2.3>

УДК 519.174

ББК 22.176

НЕЙРОСЕТЕВОЙ АНАЛИЗ РАСКРАШЕННЫХ ГРАФОВ

Илья Васильевич Гермашев

Доктор технических наук,
профессор кафедры фундаментальной информатики и оптимального управления,
Волгоградский государственный университет
germasheviv@mail.ru, fiou@volsu.ru
просп. Университетский, 100, 400062 г. Волгоград, Российская Федерация

Евгения Вячеславовна Дербишер

Кандидат технических наук,
доцент кафедры аналитической, физической химии и физико-химии полимеров,
Волгоградский государственный технический университет
derbisher1@yandex.ru
просп. В. И. Ленина, 28, 400005 г. Волгоград, Российская Федерация

Вячеслав Евгеньевич Дербишер

Доктор химических наук,
профессор кафедры технологии высокомолекулярных и волокнистых материалов,
Волгоградский государственный технический университет
derbisher-28091945@yandex.ru
просп. В. И. Ленина, 28, 400005 г. Волгоград, Российская Федерация

Елена Александровна Маркушевская

Кандидат педагогических наук,
доцент кафедры педагогики и психологии начального образования,
Волгоградский государственный социально-педагогический университет
emarkushevskaya@mail.ru
просп. В. И. Ленина, 27, 400005 г. Волгоград, Российская Федерация

Аннотация. Рассмотрена проблема идентификации раскрашенного графа, возникающая при решении задач в предметной области, формализуемой в терминах теории графов, в том числе в связи с исследованием зависимости «химическая структура – свойство». Предложена модель химической структуры в виде раскрашенного графа.

Анализ графов основан на статистиках простых цепей, для получения которых описан алгоритм поиска в ширину, приведен анализ алгоритма. Получена верхняя оценка сложности алгоритма и показана ее достижимость. Показано, как с помощью выявленных простых цепей проводится обучение искусственной нейронной сети. Формально представлен результат ее обучения в виде формулы, позволяющей вычислять выходной сигнал по вектору входных сигналов. Показано использование искусственной нейронной сети для идентификации графов. Полученные результаты представляют собой математическое обеспечение, позволяющее обоснованно создавать решающие правила идентификации систем, формализуемых в терминах теории графов.

Ключевые слова: идентификация, простая цепь, статистика, поиск в ширину, анализ алгоритма, обучение искусственной нейронной сети.

Современные методы дискретной математики, в частности теория графов, находят широкое применение при исследованиях в самых разных областях науки и техники. Так, например, в области химии, применительно к тематике наших интересов [3; 5], структуры химических соединений моделируют с помощью графов, что позволяет применять указанные математические методы для анализа и синтеза виртуальных химических структур [1; 10], имеющих не только академическое, но и практическое значение, например, при выборе перспективных веществ для продвижения на технологический рынок. При этом решение ряда важных прикладных задач в данной области часто сводится к статистическому анализу различных дескрипторов химических структур. Учитывая это, математическое решение такого рода задач состоит из четырех этапов: формального представления исходных данных (в нашем случае химических структур), построения математической модели, анализа формализованной модели и синтеза решающих правил.

Формально описанную выше задачу можно представить следующим образом.

Пусть задан универсум G графов. Выделим в нем подмножество $G^+ \subset G$ графов, обладающих интересующим нас свойством. Соответственно множество $G \setminus G^+$ обозначим через G^- . Будем считать, что, вообще говоря, элементы множества G^+ не известны, но некоторые из них все же установлены и составляют множество $G^+ \subset G^+$. Аналогично $G^- \subset G^-$. Таким образом, множество $G = G^+ \cup G^-$ составляет обучающую выборку.

Пусть задан граф $G^0 \subset G$. Требуется определить, какому из двух множеств (G^+ или G^-) он принадлежит. Иными словами, обладает ли граф G^0 заданным свойством или нет.

Таким образом, основной задачей данной работы является создание теоретических предпосылок для получения статистик графов и формирования правил идентификации свойств графов согласно обучающей выборке. Для иллюстрации и интерпретации формальных выкладок будем использовать химические структуры.

Для решения поставленной задачи будем использовать основные положения теории графов [7].

В настоящей работе для анализа графа будем использовать статистику простых цепей и при моделировании химической структуры примем следующие положения:

- каждый атом (или группу атомов) будем представлять вершиной, окрашенной в цвет, соответствующий химическому элементу (или функциональной группе);
- количество цветов вершин обозначим через k_v (k_v может достигать нескольких тысяч: химические элементы плюс группы элементов);
- для идентификации атомов в молекуле пометим все вершины в графе: v_1, \dots, v_n , где n – количество вершин в графе;
- цвет вершины v_i обозначим через c_i , $i = 1, \dots, n$;
- химические связи между атомами в молекуле будем представлять ребрами, причем ребра будут раскрашены в цвета, соответствующие различным типам химических связей. Количество цветов ребер обозначим через k_e (k_e может достигать нескольких единиц, в зависимости от учитываемых типов химической связи). Цвет ребра инцидентного вершинам v_i и v_j обозначим c_{ij} (цвет 0 обозначает отсутствие ребра).

В результате получим простой (без петель и кратных ребер) помеченный граф G как с вершинной, так и с реберной раскраской. Причем, степень любой вершины не превышает максимальной валентности химического элемента – 8, то есть

$$\Delta(G) \leq 8, \quad (1)$$

где $\Delta(G)$ – максимальная степень вершины графа G [7].

Возможны и другие ограничения, определяемые исходя из задач анализа рассматриваемого конкретного класса химических соединений. Например, если в этом же контексте рассматривать предпочтения авторов – мономеры, предназначенные для синтеза или модификации высокомолекулярных соединений [2; 4; 9; 10], то количество вершин в графе можно ограничить несколькими десятками.

Следующая задача – это определение всех возможных простых цепей в графе G (далее, поскольку речь будет идти только о простых цепях, для краткости говорится «цепь» вместо «простая цепь»).

Для решения этой задачи воспользуемся поиском в ширину [7].

1. Определим множество $H = \emptyset$ цепей, множество $V_0 = \emptyset$ рассмотренных начальных вершин цепей и $V_n = VG$ еще не рассмотренных начальных вершин.
2. Зафиксируем начальную вершину $v_{i0} \in VG \setminus V_0$ цепи, положим $V_0 = V_0 \cup \{v_{i0}\}$, $V_n = V_n \setminus \{v_{i0}\}$, множества $H_k = \emptyset$ цепей длины $k = 0, \dots, n$ с начальной вершиной v_{i0} .
3. Положим рассматриваемые цепь $h_c = v_{i0}$ и вершину $v_c = v_{i0}$, множество $V_h = \{v_{i0}\}$ вершин цепи h_c , $k = 1$.
4. Определим множество $V_c \subset VG \setminus V_h$ вершин, смежных вершине v_c . Если $V_c = \emptyset$, то перейдем на шаг 6.
5. Положим $H_k = H_k \cup \{h_c.v | v \in V_c\}$ и $H = H \cup \{h_c.v | v \in V_c\}$, где «.» – операция конкатенации.
6. Если $H_{k-1} = \emptyset$, то перейдем на шаг 7, иначе возьмем $h_c \in H_{k-1}$, в качестве v_c последнюю вершину из цепи h_c , а множества V_h все вершины цепи h_c , положим $H_{k-1} = H_{k-1} \setminus \{h_c\}$ и перейдем на шаг 4.
7. Положим $k = k + 1$. Если $k < n + 1$, то переходим на шаг 6.
8. Если $V_n = \emptyset$, то завершим алгоритм, иначе перейдем на шаг 2.

После завершения алгоритма множество H будет содержать все цепи. Заметим также, что множество H_0 цепей нулевой длины всегда пусто, поскольку такие цепи нас не интересуют.

Как видно, раскраска никак не учитывалась в алгоритме. Поэтому алгоритм применим к более широкому классу графов (и к несвязным тоже). Раскраска же нам понадобится дальше, при статистическом анализе цепей.

Покажем теперь, что приведенный алгоритм действительно находит все цепи в графе.

Рассмотрим в графе G некоторую цепь $v_{i_1}v_{i_2} \dots v_{i_m}$, где $2 \leq m \leq n$. Согласно шагам 2 и 8 вершина v_{i_1} обязательно станет начальной в цепи, а по шагу 3 вершина v_{i_1} станет рассматриваемой. Далее, поскольку v_{i_2} смежна v_{i_1} , то v_{i_2} на шаге 4 будет помещено в множество V_c и затем на шаге 5 цепь $v_{i_1}v_{i_2}$ будет помещена в множества H_1 и H . Затем по шагу 6 цепь $v_{i_1}v_{i_2}$ и вершина v_{i_2} станут рассматриваемыми. Далее повторяем приведенные выше рассуждения для связки вершин $v_{i_2} - v_{i_3}$ и цепь $v_{i_1}v_{i_2}v_{i_3}$ будет помещена во множество H .

Продолжая таким образом, добьемся того, что цепь $v_{i_1}v_{i_2} \dots v_{i_m}$ будет помещена во множество H . Поскольку цепь $v_{i_1}v_{i_2} \dots v_{i_m}$ выбрана произвольно, то это означает, что в результате работы алгоритма во множество H будут помещены все цепи графа G .

Поскольку данный алгоритм решает поставленную задачу, рассмотрим теперь, насколько эффективно это решение.

Оценим сложность L алгоритма.

Шаг 1 выполняется всего один раз и его составляют инициализация пустых множеств H, V_0 (сложность константа c_0), а также множества V_n (сложность c_1n). Шаги со 2-го по 8-й выполня-

ются, пока в качестве начальной вершины цепи не участвует каждая из вершин графа G , то есть n раз:

$$L = c_0 + c_1n + nL_1, \quad (2)$$

где L_1 – сложность шагов 2–8.

В свою очередь L_1 составляют шаг 8 (сложность c_2) и шаги со 2-го по 7-й (сложность L_2). Шаги со 2-го по 7-й повторяются, пока не рассмотрим все длины цепей k , то есть n раз. Получили, что $L_1 = c_2 + nL_2$. Подставляя полученное равенство в (2), получим

$$L = c_0 + (c_1 + c_2)n + n^2L_2. \quad (3)$$

Далее L_2 составляют шаг 7 (сложность c_3), шаг 2, на котором инициализируются H_0, \dots, H_n , и проводятся несколько операций (сложность $c_4 + c_5n$), шаг 3, на котором строим множество V_h из k вершин, учитывая, что $k \leq n$, плюс еще несколько операций, получим сложность шага 3 не больше $c_6 + c_7n$, и шаги с 4-го по 6-й (сложность L_3). Шаги с 4-го по 6-й повторяются, пока не рассмотрим все цепи из множества H_k (число таких цепей обозначим N_k). Получили, что $L_2 \leq c_3 + c_4 + c_6 + (c_5 + c_7)n + NL_3$, где $N = \max_{k=1, \dots, n} N_k$. Подставляя полученное неравенство в (3), получим

$$L \leq c_0 + (c_1 + c_2)n + (c_3 + c_4 + c_6)n^2 + (c_5 + c_7)n^3 + n^2NL_3. \quad (4)$$

На шаге 4 реализуется несколько операций и поиск смежных вершин. Все смежные вершины можно найти не более, чем за n операций. Поэтому сложность этого шага можно оценить, как не превосходящую величину $c_8 + c_9n$.

На шаге 5 добавляем в множества H_k и H новые цепи. Число этих цепей равно числу смежных вершин, а число смежных вершин не может быть больше 8 для рассматриваемого класса графов G , то есть сложность этого шага составляет константу c_{10} .

На шаге 6 реализуется несколько операций и формирование множества V_h , на что может понадобиться до n операций. Следовательно, сложность этого шага можно оценить величиной $c_{11} + c_{12}n$.

Суммируя вышесказанное, получаем $L_3 \leq c_8 + c_{10} + c_{11} + (c_9 + c_{12})n$ и подставляем это неравенство в (4):

$$\begin{aligned} L \leq c_0 + (c_1 + c_2)n + (c_3 + c_4 + c_6)n^2 + (c_5 + c_7)n^3 + (c_8 + c_{10} + c_{11})n^2N + \\ + (c_9 + c_{12})n^3N \leq c_{13}n^3N. \end{aligned} \quad (5)$$

Из этой оценки уже видно, что сложность алгоритма полиномиальна относительно числа вершин и числа цепей, что позволяет сделать заключение о достаточной эффективности алгоритма.

Теперь осталось оценить величину N .

Очевидно, что $N_1 \leq \Delta(G)$. Далее, так как одна смежная вершина – это та, из которой мы попали в следующую, то смежных вершин, отсутствующих в цепи, будет не более $\Delta(G) - 1$, и получаем $N_k \leq \Delta(G)(\Delta(G) - 1)^{k-1}$. Откуда, учитывая (1),

$$N = \max_{k=0, n} N_k \leq \Delta(G)(\Delta(G) - 1)^{n-1} = \frac{\Delta(G)}{\Delta(G) - 1} (\Delta(G) - 1)^n \leq c_{14}7^n.$$

Подставляя эту оценку в (5), получим

$$L \leq cn^37^n.$$

Очевидно, что общее количество цепей в графе не превышает величину $n(\Delta(G))^n$. Убедимся, что среди рассматриваемых графов действительно есть такие, у которых имеется экспоненциальное число цепей. Таким примером может служить, например, следующий граф порядка n на рисунке 1.

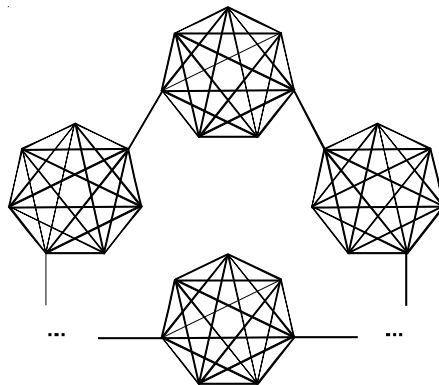


Рис. 1. Пример графа с экспоненциальным числом цепей

Пусть n будет кратно 7. Тогда возьмем ряд 7-клик (всего их будет $n/7$), последовательно соединим их ребрами (мостами) так, чтобы эти ребра не были смежными, и крайние клики соединим ребром аналогично. В каждой 7-й клике можно выделить $\sum_{k=0}^5 A_5^k$ цепей длины от 1 до 6, начинающихся с вершины, инцидентной одному мосту, и заканчивающихся вершиной, инцидентной другому мосту. Соединяя цепи из соседних клик, можно составить всего $\left(\sum_{k=0}^5 A_5^k\right)^{\frac{n}{7}}$, причем мы учли не все цепи, поэтому это нижняя оценка числа цепей графа. Учитываем также, что $\sum_{k=0}^5 A_5^k \geq \sum_{k=0}^7 C_7^k$ (в этом можно убедиться путем непосредственного вычисления). Следовательно, общее число цепей будет превосходить следующую величину

$$\left(\sum_{k=0}^5 A_5^k\right)^{\frac{n}{7}} \geq \left(\sum_{k=0}^7 C_7^k\right)^{\frac{n}{7}} = (2^7)^{\frac{n}{7}} = 2^n.$$

То есть в общем случае построить полиномиальный алгоритм для решения данной задачи, в принципе, невозможно, и предлагаемое решение является эффективным.

Таким образом, мы рассмотрели алгоритм поиска всех цепей в графе и теперь можно перейти собственно к решению задачи идентификации графа.

Пусть имеется два класса раскрашенных графов G^+ и G^- . Предъявлен некоторый граф $G^0 \notin G^+ \cup G^-$. Требуется определить, к какому из двух классов следует отнести граф G^0 .

Собственно, решающее правило будем формировать на основе полученных цепей. Для анализа статистики цепей предлагаем воспользоваться ИНС. Основные идеи ИНС для анализа химических систем в литературе представлены для различных вариантов [6; 8]. Предлагаемый подход позволяет не просто учитывать появление тех или иных цепей, но и то, какие фрагменты включают эти цепи и в каком порядке расположены фрагменты. Остановимся на формальной стороне вопроса.

Пусть $G^+ = \{G_1^+, \dots, G_{s^+}^+\}$, $G^- = \{G_1^-, \dots, G_{s^-}^-\}$ и пусть для каждого графа G_l^+ , $l = 1, \dots, s^+$, G_l^- , $l = 1, \dots, s^-$ и G_0 , получено соответственно множество H_l^+ , H_l^- и H^0 всех его цепей и $H^+ = \bigcup_{l=1}^{s^+} H_l^+$, $H^- = \bigcup_{l=1}^{s^-} H_l^-$ и $H = H^+ \cup H^-$.

Будем строить ИНС из m слоев, где m – максимальная длина из всех полученных цепей. Принцип работы ИНС следующий. На входные синапсы будем подавать цепи графа G^0 , а на выходе будем считывать сигналы. Если сумма этих сигналов положительна, то будем полагать, что $G^0 \in G^+$, если же отрицательная, то $G^0 \in G^-$.

Поскольку на входы ИНС будем подавать цепи, то примем, что нейрон соответствует вершине, а синапс – ребру. При этом число нейронов в каждом слое будет одинаковым и равно числу всех цветов вершин (плюс один для цвета ноль), встречающихся среди графов из $G = G^+ \cup G^-$. Все нейроны в одном слое окрашены в различные цвета (в том числе и в цвет ноль).

Пусть каждый из m слоев ИНС состоит из $k_v + 1$ нейронов u_p^j , где $p = 0, \dots, k_v$ (нейрон с номером $p = 0$ является фиктивным и служит для решения некоторых проблем при прохождении сигнала через ИНС) – цвет нейрона, $j = 1, \dots, m$. Условимся также считать, что любая пара нейронов, расположенная в двух соседних слоях, соединена k_e синапсами (каждый синапс соответствует ребру определенного цвета) с весом ω_{pqr}^j – вес синапса (для ребра цвета r) между нейронами u_p^j и u_q^{j+1} (рис. 2).

Вес ω_{pqr}^j формируется при обучении ИНС следующим образом. В начале обучения $\omega_{pqr}^j = 0$ для всех p, q, r и j . Рассмотрим некоторую цепь $h_{i_t}^+ = (u_{i_t^1} \dots u_{i_t^{a_t}}) \in H_l^+$, где a – длина цепи; t – номер цепи. Сигнал для $h_{i_t}^+$ подается на вход нейрона $u_{c_{i_t^1}}^1$ (имеющего тот же цвет, что и первая вершина в цепи $h_{i_t}^+$). Далее сигнал идет в следующий слой на нейрон $u_{c_{i_t^2}}^2$ по синапсу цвета $c_{i_t^1 i_t^2}$ и так далее от слоя к слою, пока не закончатся вершины в цепи $h_{i_t}^+$. При прохождении сигнала от нейрона u_p^j к нейрону u_q^{j+1} по синапсу цвета r будем увеличивать его вес ω_{pqr}^j на 1.

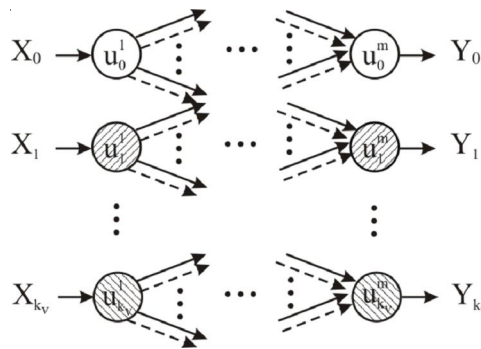


Рис. 2. Принципиальная схема ИНС:

X – входной сигнал; Y – выходной сигнал; u – нейрон; m – количество слоев ИНС;
 k – количество цветов вершины исходного графа

Так мы поступаем для каждой цепи из H_l^+ . Далее повторим эту процедуру для каждого $l = 1, \dots, s^+$.

Аналогично поступим и с цепями из множеств $H_l^-, l = 1, \dots, s^-$, только с тем отличием, что при прохождении сигнала будем вес ω_{pqr}^j синапса не увеличивать, а уменьшать на 1.

В результате получили решающее правило в виде нейронной сети, которое идентифицирует предъявляемый граф G^0 следующим образом.

Подаем на входы ИНС все цепи из H^0 . На входе каждый сигнал имеет величину 0. Сигнал для каждой цепи проходит через ИНС так, как это было описано выше при обучении сети, но теперь при прохождении сигнала по синапсу происходит не модификация его веса ω_{pqr}^j , а считывание. Считанное значение добавляется к величине сигнала.

Если длина a_{0_t} цепи оказывается меньше m , то вслед за последней вершиной этой цепи сигнал идет на фиктивный нейрон $u_{0_t^{a_{0_t}+1}}$ и далее сигнал идет через фиктивные нейроны вплоть до u_0^m . А поскольку веса синапсов между фиктивными нейронами нулевые, то сигнал до самого выхода не изменяется.

Обозначим через u_t получившуюся величину сигнала на выходе из сигнала $X_{c_{i_t^1}}$ на входе ИНС для цепи $h_{i_t}^+ = (u_{i_t^1} \dots u_{i_t^{a_t}}) \in H^0$. Тогда

$$y_t = \sum_{j=1}^{a_t-1} \omega_{c_{tj}^j c_{tj+1}^j c_{tj+1}^j}.$$

Суммарный же сигнал Y на выходе ИНС будет следующим:

$$Y = \sum_{p=0}^{k_p} Y_p = \sum_{t=1}^{|H^0|} y_t = \sum_{t=1}^{|H^0|} \sum_{j=1}^{a_t-1} \omega_{c_{tj}^j c_{tj+1}^j c_{tj+1}^j},$$

где Y_p – величина выходного сигнала.

В результате, если $Y > 0$, то идентифицируем граф G^0 , как G^+ . Если же $Y < 0$, то идентифицируем граф G^0 , как G^- .

Предложенная здесь теория может быть реализована в нейрокомпьютерной технологии для решения задач проектирования путем проведения статистического анализа простых раскрашенных графов. Естественно, приведенные выше выкладки требуют определенной адаптации для решения конкретных задач. Сказанное выше следует рассматривать, как математическое обеспечение, на основе которого можно обоснованно синтезировать конкретные решающие правила, в том числе путем введения в ИНС активационных функций, персептронов или, быть может, других компонентов, которые позволят эффективно обучать ИНС для решения конкретных задач анализа и синтеза графовых структур.

СПИСОК ЛИТЕРАТУРЫ

1. Батыршин, И. З. К анализу предпочтений в системах принятия решений / И. З. Батыршин // Тр. Моск. энергет. ин-та. – 1981. – Вып. 533. – С. 57–62.
2. Гермашев, И. В. Вычислительное прогнозирование и проектирование веществ / И. В. Гермашев, В. Е. Дербишер. – Saarbrücken (Germany) : LAPLAMBERT Academic Publishing GmbH & Co. KG, 2012. – 268 с.
3. Гермашев, И. В. Решение задач в химической технологии средствами нечетких множеств / И. В. Гермашев, В. Е. Дербишер. – Волгоград : Перемена, 2008. – 143 с.
4. Дербишер, Е. В. Прогнозирование класса опасности веществ на основе выборочных данных об их физико-химических и медико-биологических свойствах : дис. ... канд. техн. наук / Дербишер Евгения Вячеславовна. – Волгоград, 2005. – 127 с.
5. Диагностика возможной активности производных адамантана в полимерных композициях методами молекулярного дизайна / В. В. Орлов, В. Е. Дербишер, Ю. Л. Зотов, П. М. Васильев, И. В. Гермашев, Е. В. Дербишер, А. Ю. Колоскова // Химическая промышленность. – 2003. – Т. 80, № 2. – С. 46–55.
6. Круглов, В. В. Искусственные нейронные сети. Теория и практика / В. В. Круглов, В. В. Борисов. – М. : Горячая линия-Телеком, 2002. – 382 с.
7. Лекции по теории графов / В. А. Емеличев, О. И. Мельников, В. И. Сарванов, Р. И. Тышкевич. – М. : Наука, 1990. – 384 с.
8. Хайкин, С. Нейронные сети. Полный курс / С. Хайкин. – М. : Вильямс, 2006. – 1104 с.
9. Computer-aided design of chemical compounds with controlled properties / I. V. Germashev, V. E. Derbisher, M. N. Tsapleva, E. V. Derbisher // Theor. Found. of Chem. Eng. – 2004. – Vol. 38, № 1. – P. 86–91.
10. Derbisher, V. E. Fuzzy-Set-based Quantitative Estimates of the Efficiency of Thermo- and Photostabilizing Additives in Polymeric Compositions / V. E. Derbisher, I. V. Germashev, G. G. Bodrova // Polymer Science. Ser. A. – 1997. – Vol. 39, № 6. – P. 630–633.

REFERENCES

1. Batyrshin I.Z. K analizu predpochteniy v sistemakh prinyatiya resheniy [The Preferences Analysis of Decision Support Systems]. *Trudy Moskovskogo energeticheskogo instituta*, 1981, iss. 533, pp. 57-62.
2. Germashev I.V., Derbisher V.E. *Vychislitelnoe prognozirovanie i proektirovanie veshchestv* [The Computational Prediction and Design of Substances]. Saarbrücken (Germany), LAPLAMBERT Academic Publishing GmbH & Co. KG, 2012. 268 p.

3. Germashev I.V., Derbisher V.E. *Reshenie zadach v khimicheskoy tekhnologii sredstvami nechetkikh mnozhestv* [The Solution of Problems in Chemical Engineering by Fuzzy Sets Tools]. Volgograd, Peremena Publ., 2008. 143 p.
4. Derbisher E.V. *Prognozirovaniye klassa opasnosti veshchestv na osnove vyborochnykh dannykh ob ikh fiziko-khimicheskikh i mediko-biologicheskikh svoystvakh : dis. ... kand. tekhn. nauk* [The Prediction of Substances Hazard Classes Based on the Selective Data of Its Physicochemical and Biomedical Properties. Cand. tech. sci. diss.]. Volgograd, 2005. 127 p.
5. Orlov V.V., Derbisher V.E., Zotov Yu.L., Vasilyev P.M., Germashev I.V., Derbisher E.V., Koloskova A.Yu. Diagnostika vozmozhnoy aktivnosti proizvodnykh adamantana v polimernykh kompozitsiyakh metodami molekulyarnogo dizayna [The Diagnosis of Possible Activity of a Derivative Adamantane in Polymeric Compositions by Molecular Design Methods]. *Khimicheskaya promyshlennost*, 2003, vol. 80, no. 2, pp. 46-55.
6. Kruglov V.V., Borisov V.V. *Iskusstvennye neyronnye seti. Teoriya i praktika* [Artificial Neural Networks. Theory and Practice]. Moscow, Goryachaya liniya-Telekom Publ., 2002. 382 p.
7. Emelichev V.A., Melnikov O.I., Sarvanov V.I., Tyshkevich R.I. *Lektsii po teorii grafov* [Lectures on Graph Theory]. Moscow, Nauka Publ., 1990. 384 p.
8. Khaykin S. *Neyronnye seti. Polnyy kurs* [Neural Networks. Full Course]. Moscow, Williams Publ., 2006. 1104 p.
9. Germashev I.V., Derbisher V.E., Tsapleva M.N., Derbisher E.V. Computer-Aided Design of Chemical Compounds With Controlled Properties. *Theoretical Foundations of Chemical Engineering*, 2008, vol. 38, no. 1, pp. 86-91.
10. Derbisher V.E., Germashev I.V., Bodrova G.G. Fuzzy-Set-Based Quantitative Estimates of the Efficiency of Thermo- and Photostabilizing Additives in Polymeric Compositions. *Polymer Science, Ser. A.*, 1997, vol. 39, no. 6, pp. 630-633.

THE NEURAL NETWORK ANALYSIS OF COLORED GRAPHS

Ilya Vasilyevich Germashev

Doctor of Technical Sciences,
Professor, Department of Fundamental Informatics and Optimal Control,
Volgograd State University
germasheviv@mail.ru, fiou@volsu.ru
Prosp. Universitetsky, 100, 400062 Volgograd, Russian Federation

Evgeniya Vyacheslavovna Derbisher

Candidate of Technical Sciences,
Associate Professor, Department of Analytical, Physical Chemistry and Physico-Chemistry of Polymers,
Volgograd State Technical University
derbisher1@yandex.ru
Prosp. Lenina, 28, 400005 Volgograd, Russian Federation

Vyacheslav Evgenyevich Derbisher

Doctor of Chemical Sciences,
Professor, Department of High-Molecular and Fibrous Materials Technology,
Volgograd State Technical University
derbisher-28091945@yandex.ru
Prosp. Lenina, 28, 400005 Volgograd, Russian Federation

Elena Aleksandrovna Markushevskaya

Candidate of Pedagogical Sciences,
Associate Professor, Department of Pedagogy and Psychology of Primary Education,
Volgograd State Social-Pedagogical University
emarkushevskaya@mail.ru
Prosp. Lenina, 27, 400005 Volgograd, Russian Federation

Abstract. The article deals with the problem of colored graph identification. This problem arises when solving tasks in the subject area which is formalized in terms of graph theory including cases connected with the investigation of “chemical structure – property” dependence. The authors propose the model of chemical structure in the form of a colored graph. The upper bound for the algorithm complexity is obtained, and its feasibility is shown. The learning samples are represented by graphs with a given property. The problem solved in the paper is the method development and analysis allowing to identify the property of graph that is not included in the learning samples. To solve the problem it is proposed to use the mechanism of artificial neural network of the original structure, the principles of which differ significantly from generally accepted, which is in the signals form and signals distribution ways across the network. The graph analysis is based on simple chains statistics for which the breadth-first search algorithm is described, and the algorithm analysis is given. The proposed algorithm allows also to handle disconnected graphs and thus to analyze multi-component systems. The article presents the formal result of its training as a formula allowing to calculate the output signal for input signals vector. The use of artificial neural network for graphs identification is demonstrated. The obtained results represent mathematical software which enables creating a reasonable decision rules for a systems identification formalized in graph theory terms.

Key words: identification, simple chain, statistics, breadth-first search, algorithm analysis, artificial neural network training.