На правах рукописи

Мальсагов Магомед Юсупович

ПРИМЕНЕНИЕ ДИСКРЕТИЗАЦИИ ДЛЯ РЕШЕНИЯ ЗАДАЧ БИНАРНОЙ ОПТИМИЗАЦИИ С ПОМОЩЬЮ НЕЙРОННОЙ СЕТИ ХОПФИЛДА

05.13.01 – Системный анализ, управление и обработка информации
(по математическим отраслям и информатике)

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата физико-математических наук

Москва - 2012

Работа выполнена в Федеральном государственном бюджетном учреждении науки Научно-исследовательском институте системных исследований РАН.

Научный руководитель: кандидат физико-математических наук
Крыжановский Михаил Владимирович

Официальные оппоненты: Литвинов Олег Станиславович,
доктор физико-математических наук, профессор,
МГТУ им. Н.Э. Баумана

Доленко Сергей Анатольевич,
кандидат физико-математических наук,
старший научный сотрудник, НИИЯФ МГУ

Ведущая организация: Национальный исследовательский ядерный университет «МИФИ»

Защита состоится «25» апреля 2012 года в 16 часов на заседании диссертационного совета Д 002.265.01 при НИИСИ РАН по адресу 117218, Москва, Нахимовский проспект, д. 36, корп. 1, конференц-зал.

С диссертацией можно ознакомиться в библиотеке НИИСИ РАН (комн. 13-21а)

Автореферат разослан «23» марта 2012 года

Ученый секретарь
диссертационного совета,
кандидат физико-математических
наук, доцент В.Б. Демидович

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. Задачи дискретной оптимизации широко распространены и встречаются практически во всех сферах человеческой деятельности. Многочисленные проблемы в математике, статистике, технике, науке, медицине и экономике могут рассматриваться как проблемы оптимизации. Задачей оптимизации является нахождение решения, которое удовлетворяет системе ограничений и максимизирует или минимизирует целевую функцию. Наиболее известными задачами дискретной оптимизации являются задача о рюкзаке, задача о коммивояжере, задача планирования вычислений для многопроцессорной системы, выбор оптимальной структуры автоматизированной системы управления и т.д. Решение этих задач связано с рядом трудностей. Например, полный перебор возможных решений, как правило, невозможен из-за большого объема вычислений.

Основными методами дискретной оптимизации являются метод ветвей и границ, динамическое программирование, метод отсечений, генетические алгоритмы, нейросетевые алгоритмы. Потребность в большой оперативной памяти и вычислительная сложность остаются слабыми местами всех известных оптимизационных алгоритмов. Так, например, метод ветвей и границ позволяет решать только задачи малой размерности $(N < 100)$. Поэтому остается актуальной задача построения и модификации алгоритмов с целью увеличения их производительности и снижения требований на вычислительные ресурсы. Очень удобными для решения этих задач оказались искусственные нейронные сети.

Искусственные нейронные сети — математические модели, а также их программные или аппаратные реализации, построенные по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма. Это понятие возникло при изучении процессов, протекающих в мозге, и при попытке смоделировать эти процессы. Нейронные сети позволяют быстро находить приближенные решения задач дискретной оптимизации больших размерностей $(N > 1000)$. Однако здесь тоже возникают проблемы с оперативной памятью и вычислительной сложностью.

Хотя искусственные нейронные сети хорошо показали себя в решении задач дискретной оптимизации, большинство проводимых исследований направлено на создание алгоритмов отыскания более глубоких минимумов, а не на увеличение их быстродействия или экономии памяти. Данная диссертационная работа ставит своей целью найти новые методы оптимизации, удовлетворяющие более жестким требованиям по времени их работы.

Цель диссертационной работы. Основная цель диссертационной работы состояла в разработке быстрого метода решения задач комбинаторной оптимизации на базе нейронных сетей.

В диссертационной работе были решены следующие задачи:

1. Предложена и исследована процедура дискретизации матричных элементов, позволяющая ускорить процесс минимизации многоэкстремального квадратичного функционала, построенного в пространстве состояний с бинарными переменными.

2. Теоретически и экспериментально получены оценки для вероятности несовпадения направлений градиентов и эффективности минимизации.
3. На основе процедуры дискретизации разработаны и исследованы алгоритмы минимизации квадратичного функционала. Исследовано увеличение быстродействия новых алгоритмов и их эффективность по сравнению со стандартной моделью Хопфилда.

Основные положения, выносимые на защиту:

1. Процедура дискретизации матричных элементов, позволяющая ускорить процесс минимизации многоэкстремального квадратичного функционала, построенного в пространстве состояний с бинарными переменными.
2. Алгоритмы поиска минимумов квадратичного функционала в пространстве бинарных переменных с использованием процедуры дискретизации.

Методы исследований. Для решения поставленных задач в работе были использованы методы вычислительной математики, теории вероятностей и математической статистики, а также методы прикладного программирования.

Научная новизна. В диссертационной работе

1. Предложена и исследована процедура дискретизации.
2. Создан алгоритм, позволяющий увеличить скорость решения оптимизационных задач и получать при существенно меньших вычислительных затратах заметно лучшее решение, чем при использовании стандартной модели Хопфилда.
3. Применение дискретизации снижает требования к оперативной памяти, что позволяет решать задачи недоступные стандартной модели Хопфилда.
4. Получены выражения, позволяющие аналитически оценить качество решения при использовании минимизационного алгоритма.

Практическая ценность. Практическая ценность результатов работы состоит в следующем:

1. На основе процедуры дискретизации создан алгоритм, позволяющий уменьшить время решения оптимизационных задач и получать при существенно меньших вычислительных затратах лучшее решение, чем при использовании стандартного подхода, основанного на модели Хопфилда;
2. Получены выражения, позволяющие априори аналитически оценить качество решения, которое может быть получено в результате минимизации.

Разработанные алгоритмы и методы дискретизации найдут свое применение в прикладных системах на основе нейронных сетей Хопфилда.

Апробация работы и публикации. По материалам диссертации опубликовано 11 работ, из них 5 – в российских и зарубежных журналах [1-5] (все из перечня ВАК), 6 – в трудах конференций [5-11].

Основные положения работы докладывались на следующих конференциях:

1. XI Всероссийская научно-техническая конференция «Нейроинформатика-2009», Москва, 2009.
2. Международная Научно-Техническая конференция «Искусственный Интеллект. Интеллектуальные Системы – 2009», ИИ – 2009, Украина, 2009.
3. XII Всероссийская научно-техническая конференция «Нейроинформатика-2010», Москва, 2010.

4. The fourth International Conference on Neural Networks and Artificial Intelligence, ICNNAI – 2010, респ. Белоруссия, Брест, 2010.
5. XIII Всероссийская научно-техническая конференция «Нейроинформатика-2011», Москва, 2011.
6. Международная Научно-Техническая конференция «Искусственный Интеллект. Интеллектуальные Системы – 2011», ИИ – 2011, Украина, 2011.

Структура и объем диссертации. Работа состоит из четырех глав, заключения, списка литературы и двух приложений. Общий объем диссертации составляет 133 страницы. Список литературы содержит 121 наименование.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

В **первой главе** представлен обзор работ посвященных исследованию нейронной сети Хопфилда и ее применению к решению задач дискретной оптимизации. Сформулированы цели диссертационной работы, обоснована ее актуальность и научная новизна.

Модель Хопфилда принято описывать как систему взаимосвязанных нейронов (рис.1). Состояние сети описывается $N-$мерным вектором $\mathbf{S}$, компоненты которого, бинарные переменные, принимают значения ± 1. Связи между нейронами задаются с помощью симметричной матрицы связей $\hat{\mathbf{A}}$ с нулевыми диагональными элементами. Каждый нейрон связан со всеми остальными.

Пусть в начальный момент времени сеть находится в состоянии $\mathbf{S_0}$. На каждый нейрон со стороны остальных нейронов действует локальное поле $\mathbf{H}$

$$\mathbf{H} = -\mathbf{B} + \hat{\mathbf{A}} \cdot \mathbf{S}. \qquad (1)$$

Под влиянием этого поля компоненты состояния $\mathbf{S}(t)$ меняются по правилу:

$$s_i(t+1) = \begin{cases} s_i(t), & s_i(t)H_i(t) \geq 0 \\ -s_i(t), & s_i(t)H_i(t) < 0 \end{cases}, \; i = 1,\ldots,N. \qquad (2)$$

Состояние нейрона не меняется, если ориентация этого нейрона и локального поля одинаковы, и меняется на противоположное в противном случае. Вектор $\mathbf{B}$ задает пороги нейронов, т.е. границу, при превышении которой нейрон меняет свое состояние.

Состояние сети характеризуется функцией энергии E

$$E = -\frac{1}{2}\left(\mathbf{S}, \hat{\mathbf{A}}\mathbf{S}\right) + \left(\mathbf{B}, \mathbf{S}\right). \qquad (3)$$

В процессе эволюции сети энергия состояния неуклонно понижается. Постепенно система приходит в состояние покоя.

Нейронные сети Хопфилда применяются для решения широкого круга задач: распознавание образов, классификация, кластеризация, прогнозирование, аппроксимация функций и др. Подавляющее число исследований направлено на изучение таких характеристик нейронных сетей как емкость памяти, помехоустойчивость и т.д.

Возможность использования нейронных сетей для задач оптимизации была впервые продемонстрирована Хопфилдом в его совместных работах с Танком.

Успешное применение нейронной сети к задаче коммивояжера инициировало исследование нейросетевых подходов к решению задач оптимизации.

Наличие большого количества локальных минимумов, которыми обладает нейронная сеть большой размерности, затрудняет решение задач дискретной оптимизации, поскольку нахождение приемлемого субоптимального состояния требует большого объема вычислений. Например, сеть Хопфилда размерности 100 имеет более 50 000 локальных минимумов, и нахождение субоптимального решения требует ~ 1 часа машинного времени. В случае размерности сети 1000 вероятность нахождения наиболее глубокого минимума составляет $\sim 10^{-6}$ и требует ~ 500 часов работы алгоритма.

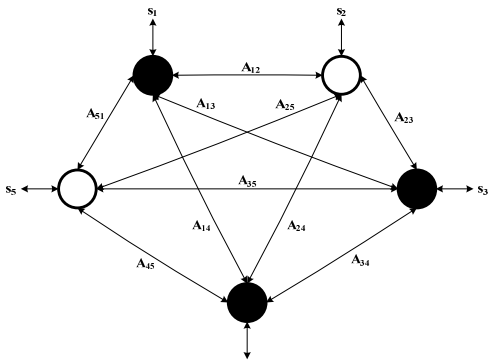

Рис.1. Нейронная сеть Хопфилда.

С целью увеличения быстродействия нейронных сетей, в 2006 году Магомедов Б.М. предложил метод снижения размерности сети, путем объединения нейронов в группы (домены). В этом подходе одновременно меняют состояние всей группы нейронов. Однако, при этом глубина находимых локальных минимумов недостаточна. Дальнейшие исследования в этом направлении прекращены.

Других работ, посвященных увеличению быстродействия нейронных сетей, нет.

В данной работе исследуется возможность применения целочисленной арифметики для решения задач бинарной оптимизации с помощью нейронной сети Хопфилда с целью увеличения быстродействия вычислительных алгоритмов.

Во **второй главе** описывается процедура дискретизации, заключающаяся в замене действительных элементов матрицы межсвязей нейронной сети целыми числами.

Решение задач дискретной оптимизации нейронными сетями сводится к минимизации квадратичного функционала (3).

Дискретизация заключается в замене матрицы $\hat{\mathbf{A}}$ некоторой матрицей $\hat{\mathbf{C}}$. Элементами матрицы $\hat{\mathbf{C}}$ являются целые числа в диапазоне $[-q; q]$, где q - число градаций, свободный параметр, задаваемый пользователем. Матрица $\hat{\mathbf{C}}$ формируется следующим образом: разбиваем область распределения элементов центрированного остатка $A'_{ij} = A_{ij} - A_0$ на $(2q+1)$ отрезка. Здесь A_0 - среднее значение матричного элемента. Формируем матрицу $\hat{\mathbf{C}}$ по правилу

$$C_{ij} = k \cdot sign\left(A'_{ij}\right), \text{ когда } x_{k-1} < \left|A'_{ij}\right| \le x_k,\ k = 0, 1, \ldots, q. \qquad (4)$$

Здесь x_k - правая граница k-ого отрезка.

Длины отрезков выбираются так, чтобы средние значения на отрезках были кратны величине m, где $m > 0$ - наименьшее положительное среднее

значение (среднее на k-ом отрезке равно km, $k = 0, \pm 1, ..., \pm q$). Заметим, что длины всех отрезков однозначно определяются длиной l_0 отрезка с нулевым средним, которая является свободным параметром для последующей оптимизации. Более того, введение отрезка с нулевым средним является ключевым моментом, позволяющим наилучшим образом аппроксимировать матрицу $\hat{\mathbf{A}}$ матрицей $\hat{\mathbf{C}}$ и существенно повысить эффективность алгоритма минимизации.

Адаптация алгоритма минимизации. Адаптируем алгоритм минимизации к работе с матрицей $\hat{\mathbf{C}}$. Для этого сделаем в (3) замены $A_{ij} \to C_0 + C_{ij}$ и $B_i \to b_i$. Здесь C_0 и b_i - свободные параметры, оптимальные значения которых будут определены ниже. Тогда, расчет локального поля и обновление компонент конфигурационного вектора будет производиться в соответствии с выражениями:

$$h_i = -b_i + \sum_{j \neq i}^{N} (C_0 + C_{ij}) s_j \ , \ s_i(t+1) = \begin{cases} s_i(t), & s_i(t) h_i(t) \geq 0 \\ -s_i(t), & s_i(t) h_i(t) < 0 \end{cases}. \qquad (5)$$

Решающее правило (5) соответствует спуску по поверхности, описываемой дискретизированным функционалом

$$\varepsilon = -\frac{1}{2}\left(\mathbf{S}, \left(C_0 + \hat{\mathbf{C}}\right)\mathbf{S}\right) + (\mathbf{b}, \mathbf{S}). \qquad (6)$$

Заменить решающее правило (2) его дискретизированным аналогом (5) можно только в том случае, когда знаки локальных полей H_i и h_i совпадают в любой точке N-мерного пространства с достаточно большой вероятностью (амплитуды полей H_i и h_i не играют роли). Иными словами, использовать модифицированное правило (5) для минимизации исходного функционала (3) можно, если свести до минимума величину ошибки – вероятность несовпадения направлений локальных полей в случайной точке пространства:

$$P = \Pr\{H_i > 0 \cap h_i < 0\} + \Pr\{H_i < 0 \cap h_i > 0\}, \qquad (7)$$

Для минимизации ошибки надо найти наилучшее представление матрицы $\hat{\mathbf{A}}'$ матрицей $\hat{\mathbf{C}}$ и найти оптимальные значения величин C_0 и b_i, $i = \overline{1, N}$.

Выражение для ошибки приведено в диссертационной работе. Оптимальные значения величин C_0 и b_i определяются из условий $\partial P / \partial b_i = 0$ и $\partial P / \partial C_0 = 0$:

$$C_0 = A_0 \frac{\sigma_C^2}{\langle C_{ij} A'_{ij} \rangle}, \quad b_i = B_i \frac{\sigma_C^2}{\langle C_{ij} A'_{ij} \rangle}. \qquad (8)$$

Анализ полученных выражений показал, что вероятность ошибки P не зависит от знаков величин A_0 и B_i. С ростом абсолютных значений $|A_0|$ и $|B_i|$ вероятность ошибки P быстро уменьшается (рис. 2).

Наихудшие условия для дискретизации имеют место, когда $A_0 = 0$ и $B_i = 0$. В этом случае величина ошибки максимальна и описывается выражением:

$$P_{\max} = \frac{1}{2} - \frac{1}{\pi} \arcsin \rho_{\min}, \quad \rho_{\min} = \frac{\langle C_{ij} A'_{ij} \rangle}{\sigma_C \sigma_A}. \qquad (9)$$

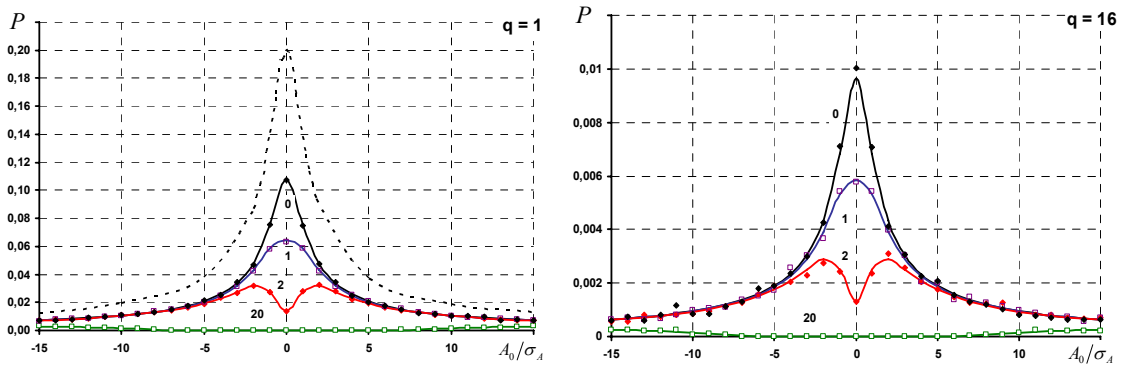

Рис.2. Вероятность несовпадения направлений локальных полей в случайной точке конфигурационного пространства. Матрицы с равномерным распределением элементов.

Отметим, что все выражения получены в самом общем виде, без привязки к конкретному распределению матричных элементов A_{ij} и типу выбранного метода дискретизации. Поэтому можно сделать общий вывод: наилучшее представление матрицы $\hat{\mathbf{A}}'$ матрицей $\hat{\mathbf{C}}$ сводится к такому подбору метода дискретизации, при котором величина $\rho_{\min}$ максимальна: согласно (9) вероятность ошибки $P_{\max}$ при этом минимальна. В этом смысле выбранный нами метод дискретизации, направленный на максимальное ускорение алгоритма минимизации, как правило, не является наилучшим.

Выражение (9) следует оптимизировать по длине отрезка с нулевым средним, находя l_0 из условия

$$\frac{\partial P}{\partial l_0} = 0\,. \qquad (10)$$

В диссертационной работе рассматривались матрицы с равномерным и нормальным распределениями матричных элементов. Для матриц с равномерным распределением при $q = 1$ получаем $P_{\max} \approx 0.11$. С ростом числа градаций до $q = 16$ вероятность ошибки снижается до $P_{\max} \approx 0.01$. Для матриц с нормальным распределением для тех же значений параметра q ошибка меняется от значения $P_{\max} \approx 0.16$ до $P_{\max} \approx 0.06$. Несмотря на то, что тип распределения влияет на эффективность дискретизации, ошибка все равно остается достаточно малой величиной.

Минимум функционала. Процесс минимизации функционала (3) начинается с некоторой случайной конфигурации $\mathbf{S}$. Подчиняясь решающему правилу (5) нейронная сеть приходит в некоторое устойчивое состояние $\mathbf{S}_0^*$, являющееся минимумом дискретизированного функционала (6). Если из этой точки продолжить спуск с решающим правилом (2), то сеть придет в состояние $\mathbf{S}_0$, соответствующее минимуму функционала (3). На всем пути $\mathbf{S} \to \mathbf{S}_0^* \to \mathbf{S}_0$ значение ошибки только уменьшается. В диссертационной работе анализируется так же величина ошибки в точке $\mathbf{S}_0$ - минимуме исходного функционала (3).

Определив вероятность ошибки с помощью выражения (9), можно сразу же оценить и расстояние (как по Хэммингу d, так и по энергии dE) между

найденным минимумом дискретизированного функционала $\mathbf{S}_0^*$ и истинным минимумом $\mathbf{S}_0$, до которого сеть не добралась. Обладая такой информацией можно принять решение: продолжать ли спуск дальше из точки $\mathbf{S}_0^*$, или же остановиться (в зависимости от требований поставленной задачи):

$$d = PN,$$
$$dE = \sigma_{\mathbf{A}} PN \sqrt{\frac{N}{2\pi}}. \qquad (11)$$

Проведенный анализ показал, что расстояние между минимумами достаточно мало: для матриц с равномерным распределением не превышает $d = 0.11N$ при $q = 1$ и снижается до значения $d = 0.02N$ при $q = 16$. Эффективность dE дискретизированного алгоритма достаточно велика: разница в энергиях составляет всего лишь 7% при $q = 1$ и меньше 0.2% при $q = 16$.

В **третьей главе** рассматриваются особенности программной реализации процедуры дискретизации. На основе идеи дискретизации разработаны и исследованы алгоритмы минимизации квадратичного функционала. Исследовано быстродействие и эффективность новых алгоритмов по сравнению со стандартной моделью Хопфилда.

При дискретизации исходная матрица $\hat{\mathbf{A}}$ заменяется ее дискретизированным аналогом $\hat{\mathbf{C}}$, элементы которой являются целыми числами

$$-q \le C_{ij} \le q, \qquad (12)$$

где q - целое число градаций. Например, при $q = 1$ - $C_{ij} = -1; 0; +1$. В этом случае нахождение состояний определяется минимизацией функционала ε, описываемого выражением (6). Динамика системы описывается выражением (5).

На рисунке 3 показы плотности распределений по энергии исходного функционала E (3) в локальных минимумах дискретизированного функционала для нескольких значений числа градаций: $q = 1, 2, 16$. Видно, что с увеличением числа градаций пик плотности распределения смещается влево, в сторону более глубоких минимумов. То есть вероятность отыскания более глубоких минимумов увеличивается. Большая часть пути (около 95%) при минимизации проходится при использовании дискретизированного функционала (рис. 4). На графиках все кривые нормированы на модуль среднего значения энергии минимумов сети Хопфилда. Таким образом, начальные состояния (кривая 1) распределены около нуля, а минимумы сети Хопфилда (кривая 3) распределены около -1.

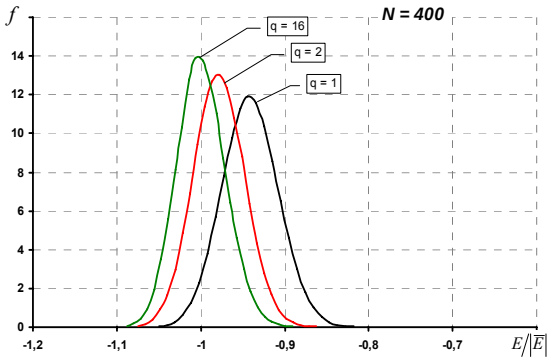

Рис.3. Плотность распределения локальных минимумов.

Минимумы дискретизированного функционала (кривая 2) по энергии отстоят от начальных состояний на 0.95.

Если использовать дискретизированные минимумы как стартовые точки исходной сети, то с большей вероятностью найдем более глубокие минимумы (кривая 4), чем при стандартной процедуре минимизации.

При оптимизации количество итераций для дискретизированной матрицы примерно равно числу итераций исходной матрицы (рис. 5). Таким образом, путь, проходимый при оптимизации по энергии примерно равен пути проходимому на исходной матрице, а объем вычислений при такой замене тот же. Однако применение целых чисел малой разрядности в процедуре дискретизации дает возможность для более экономного хранения этих чисел и увеличивает скорость работы алгоритма. В случае $q = 15$ в 4 байта можно записать одновременно 4 целых числа. При этом время работы сократится в 4 раза. А при $q = 1$ в 4 байта можно записать сразу 8 чисел, и время работы сократится до 8 раз. Экономия происходит за счет операций процессор-память, т.к. при использовании чисел малой разрядности можно оперировать сразу несколькими числами.

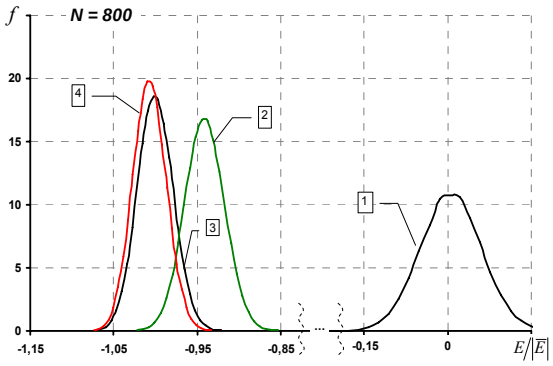

Рис.4. Плотность распределения состояний по энергии: кривая 1 – стартовые конфигурации $\mathbf{s}$; кривая 2 – локальные минимумы $\mathbf{s}_0^*$ дискретизированного функционала $(q = 1)$; кривая 3 – локальные минимумы, полученные без использования дискретизации; кривая 4 – окончательные локальные минимумы $\mathbf{s}_0$.

На рисунке 6 показано увеличение скорости алгоритма при использовании дискретизации ($q = 1$, в 4 байта записывалось 8 чисел) по сравнению с алгоритмом Хопфилда. С увеличением размерности сети скорость алгоритма увеличивается и достигает своего предельного значения 7.3.

На основе описанных выше преимуществ, предлагается два алгоритма поиска: одноэтапный и двухэтапный.

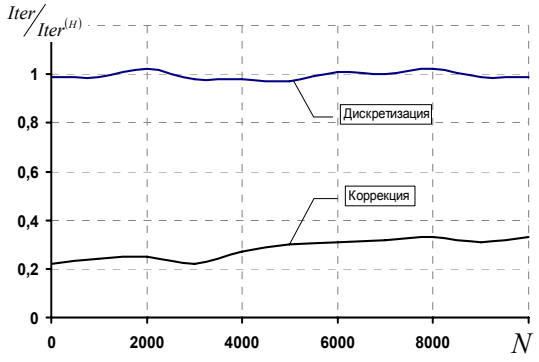

Рис.5. Количество итераций при замене вещественной матрицы на целочисленную (q=1).

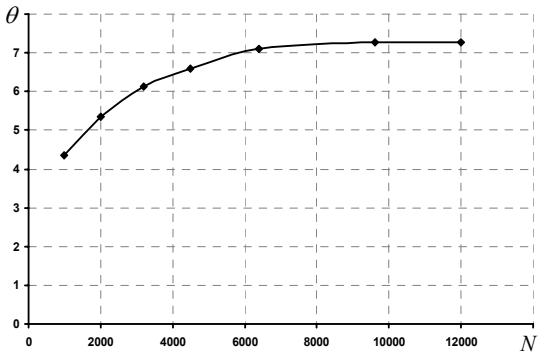

Рис.6. Ускорение алгоритма при использовании упаковки чисел малой разрядности.

Одноэтапный алгоритм поиска состоит в том, что минимизируется только дискретизированный функционал. Этот подход основан на следующей закономерности: более глубокому минимуму дискретизированного функционала $\varepsilon = \varepsilon\left(\mathbf{S}_\mathbf{0}^*\right)$ соответствует более глубокая энергия исходного функционала $E = E\left(\mathbf{S}_\mathbf{0}^*\right)$, где $\mathbf{S}_\mathbf{0}^*$ - минимум дискретизированного функционала. Сказанное выше показано на рисунке 7. По оси абсцисс отложено расстояние по энергии, на котором расположен локальный минимум относительно наименьшего значения функционала $\varepsilon = \varepsilon\left(\mathbf{S}_\mathbf{0}^*\right)$. По оси ординат отложено расстояние тех же конфигураций, но по энергии исходного функционала $E = E\left(\mathbf{S}_\mathbf{0}^*\right)$.

Оценить стандартное отклонение σ_ε и среднее значение энергии минимумов дискретизированного функционала можно по формулам

$$\bar{\varepsilon} = -\frac{\sigma_\mathbf{C} N^2}{4\sqrt{0.14N}},\ \sigma_\varepsilon = \frac{\sigma_\mathbf{C} N \sqrt{2\left(1-\bar{\varepsilon}^2/N^4\right)}}{4}. \qquad (13)$$

Видно, что практически на всем диапазоне зависимость линейная. Однако на расстоянии менее 0.05 от наименьшего значения (около $1.5\sigma_\varepsilon$ от среднего значения энергии минимумов) эта зависимость нарушается. Поэтому нельзя утверждать, что, отыскав самый глубокий дискретизированный минимум, мы наилучшим образом минимизировали исходный функционал.

С учетом сказанного, одноэтапный алгоритм можно сформулировать следующим образом: после дискретизации матрицы определяются минимумы дискретизированного функционала, лежащие глубже $1.5\sigma_\varepsilon$ от среднего значения $\bar{\varepsilon}$. Затем, в зависимости от цели задачи, можно либо ограничиться самым глубоким минимумом дискретизированного функционала, либо вычислить соответствующие им энергии исходного функционала и выбрать

самое глубокое состояние из них. Второй способ более медленный, так как вычисление энергии исходного функционала требует $\sim 2N^2$ операций. Вычисление энергии дискретизированного функционала может быть выполнено быстро суммированием локальных полей (уже известных на последнем этапе минимизации) с соответствующими знаками. Так же необходимо отметить, что минимумы дискретизированного функционала не являются минимумами исходного функционала. Поэтому одноэтапный алгоритм может быстро минимизировать функционал, но не гарантирует отыскание минимума исходного функционала.

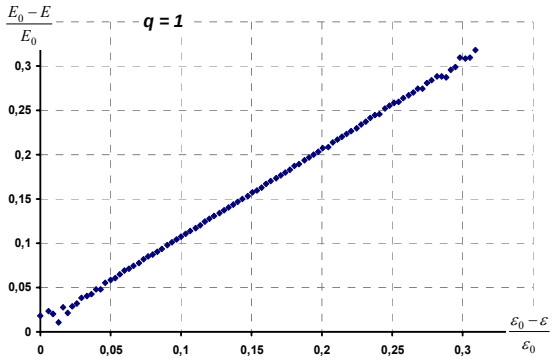

Рис.7. Соотношение энергий минимумов одноэтапного алгоритма на исходном функционале и дискретизированном функционале.

Двухэтапный алгоритм. Для нахождения минимумов исходного функционала предлагается двухэтапный алгоритм: после нахождения минимумов дискретизированного функционала, они используются как стартовые состояния сети Хопфилда на исходной матрице. Однако при таком подходе теряется быстродействие алгоритма, достигаемое дискретизацией. Поэтому предлагается производить второй этап лишь с части состояний, найденных на первом этапе. Конечно, при таком подходе будут потери в эффективности минимизации. Эффективность минимизации оценивается выражением

$$\delta E = \frac{E_0 - E^*}{E_0}, \qquad (14)$$

где E_0 - самый глубокий минимум, найденный в результате стартов со всех состояний $\mathbf{S_0^*}$, E^* - самый глубокий минимум, найденный в результате стартов из заданной области.

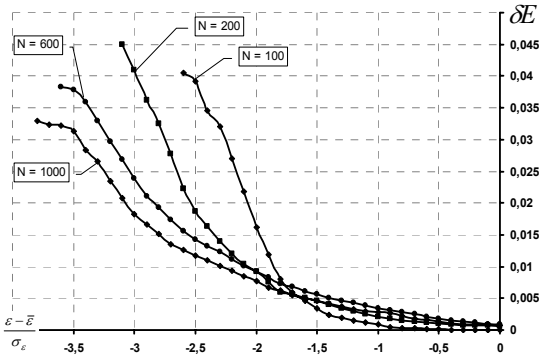

Рис.8. Зависимость эффективного расстояния от размера стартовой области.

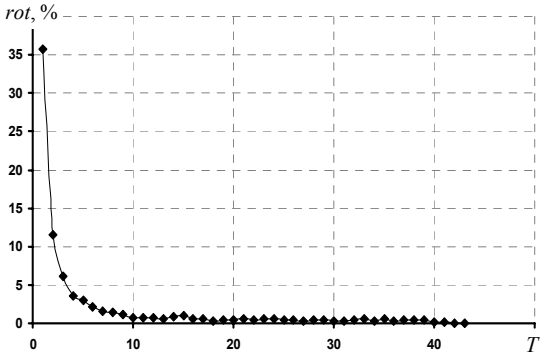

Рис.9. Количество изменений состояний нейронов на каждой итерации. Размерность нейросети $N = 4000$, T - номер итерации.

На рисунке 8 показано, как меняется эффективность минимизации при уменьшении числа минимумов дискретизированного функционала, используемых на втором этапе. По оси абсцисс отложена верхняя граница минимумов дискретизированного функционала, которые используются как стартовые конфигурации на втором этапе. По оси ординат отложена эффективность минимизации δE. Видно, что если сдвинуть границу на $\approx 2\sigma_\varepsilon$ относительно среднего значения энергии минимумов дискретизированного функционала, то эффективность алгоритма снизится приблизительно на 1%. При этом число состояний, подлежащих коррекции на втором этапе, составит порядка 0.05 от общего числа состояний $\mathbf{S}_\mathbf{0}^*$, найденных на первом этапе. Так же можно отметить, что с ростом размерности эта граница отодвигается влево, уменьшая необходимое число стартов на втором этапе.

Модификация динамики. С увеличением числа итераций, число нейронов, которые изменяют свое направление, неуклонно уменьшается (рис. 9). Это означает, что направления компонент локального поля **H** также реже изменяется уже после 4-ой итерации (менее 5%). Поэтому вычисление на каждом шаге компоненты вектора **H** не эффективно. В настоящее время

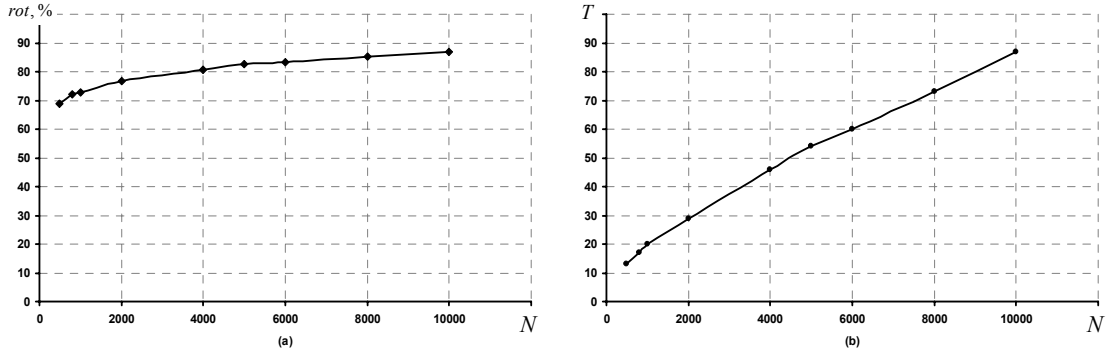

Рис.10. (a) – Полное число изменений состояний нейронов при увеличении размерности нейронной сети N; (b) – Изменение числа итераций в зависимости от размерности сети.

используется другой метод расчета. В исходном состоянии $\mathbf{S}$ вычисляются все компоненты $\mathbf{H}$. На каждом шаге процедуры при изменении состояния нейрона вектор $\mathbf{H}$ модифицируется по правилу $\mathbf{H} = \mathbf{H} \pm 2(\mathbf{A})_i$, если направление спина положительно/отрицательно, $(\mathbf{A})_i$ – i-ый вектор-столбец матрицы $\hat{\mathbf{A}}$.

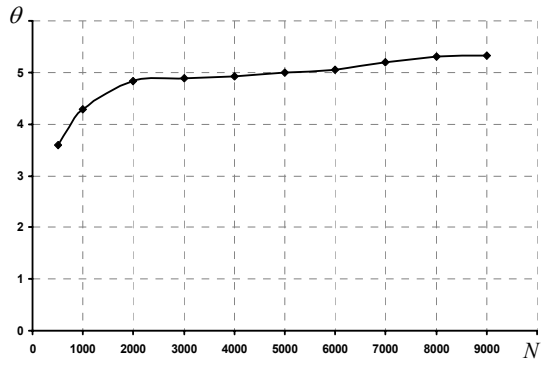

Рис.11. Ускорение, получаемое при упаковке чисел $(p = 8)$ по сравнению с обычным методом.

При перевороте каждого спина производится N операций. Число изменений состояний нейронов не превышает N (рис. 10a), поэтому вычислительная сложность такого алгоритма $\sim (1+\alpha)N^2$, а $0.5 < \alpha < 1$. Программная реализация стандартной модели Хопфилда требует порядка $Op \approx 2TN^2$ операций. Отношение объемов вычислений исходного метода к методу с обновлениями составляет величину порядка числа итераций $\sim T, (T >> 1)$ (рис.10b). Поэтому в дальнейшем использовался алгоритм с обновлениями.

Увеличение быстродействия одноэтапного алгоритма при использовании дискретизации в динамике с обновлениями для различной размерности показано на рисунке 11. Установлено, что достигаемое ускорение при упаковке 8 чисел в 4 байта ($q = 1$, $p = 8$) равно $\theta \approx 5.3$. При упаковке 4 чисел $(p = 4)$ в исходный формат ускорение составляет $\theta \approx 3.8$ $(q < 16)$. Таким образом, с помощью дискретизации можно добиться 8-кратного увеличения скорости работы алгоритма при использовании чисел малой разрядности $\{-1; 0; +1\}$.

Объем вычислений двухэтапного алгоритма складывается из объема вычислений на первом и втором этапах. Согласно рис. 5 число итераций на первом этапе равно числу итераций обычного алгоритма O_H, однако, за счет дискретизации эти операции выполняются в θ раз быстрее. Таким образом, объем вычислений на первом этапе по сравнению с алгоритмом Хопфилда составит

$$O_1 = \frac{O_H}{\theta}. \qquad (15)$$

На втором этапе производится лишь часть пусков, произведенных на первом этапе. В среднем доля пусков составляет 0.05 от общего числа стартов. При этом на втором этапе (коррекции) производится лишь 0.3 от числа итераций стандартного алгоритма Хопфилда (рис.5, нижняя кривая). Поэтому объем вычислений на втором этапе составит

$$O_2 = 0.05 \cdot 0.3 O_H. \qquad (16)$$

Тогда ускорение двухэтапного алгоритма составит

$$\Theta = \frac{O_H}{0.05 \cdot 0.3 O_H + O_H / \theta}. \qquad (17)$$

В случае $q = 1,\ p = 8 - \Theta = 5$, а при $q = 1,\ p = 16 - \Theta = 7.14$.

Таким образом, уменьшив число состояний, участвующих на втором этапе, удалось сохранить быстродействие алгоритма, при этом незначительно пожертвовав качеством минимизации.

В **четвертой главе** представлены схемы алгоритмов и анализ их быстродействия. Рассматриваются изменения в сети Хопфилда, связанные с применением дискретизации. Приводится описание программной реализации сети Хопфилда с дискретизированной матрицей.

Основным результатом четвертой главы является оценка соотношения числа операций алгоритма с упаковкой данных по сравнению с обычной сетью Хопфилда. Рассмотрим случай, когда под элемент исходной матрицы выделяется 4 байта. В зависимости от числа градаций под дискретизированный элемент можно выбирать различное число бит, желательно, с запасом, чтобы переполнение не происходило достаточно часто. Например, можно выделять $r = 4$ бита, когда $q \in \{1; 2\}$ и упаковывать $p = 8$ дискретизированных чисел в 4 байта или $r = 8$ бит, когда $q \in [3; 16]$ и упаковывать $p = 4$ дискретизированных числа в 4 байта. Тогда можно будет просуммировать v чисел разрядности r, прежде чем произойдет переполнение.

$$v = \left\lfloor \frac{2^{r-1}}{q} - 1 \right\rfloor. \qquad (18)$$

Здесь $\lfloor x \rfloor$ - наибольшее целое, меньшее или равное x.

С учетом сказанного выше можно оценить число операций, которое затратит нейронная сеть с упакованными данными на спуск из случайного состояния в минимум

$$O_p = 4NT + N\left[\frac{2}{p} + \frac{1}{2v}\right](N + rot). \qquad (19)$$

Стандартная сеть Хопфилда производит спуск из стартового состояния в минимум за число операций равное

$$O_H = 2N^2 + NT + N \cdot rot. \qquad (20)$$

В результате многочисленных экспериментов была установлена зависимости между числом итераций T, числом изменений нейронами своего состояния rot и размерностью задачи N:

$$T \approx 0.29 N^{0.62}, \quad rot \approx 0.32 N(1 + 0.2 Ln(N)). \qquad (21)$$

С учетом приведенных выше выражений можно оценить выигрыш по числу операций. На рисунке 12 представлено отношение числа операций алгоритма Хопфилда и алгоритма с упаковкой дискретизированных чисел. Видно, что за счет упаковки данных (матрицы и состояния) экономия операций составляет около 4.5 раз. С увеличением числа градаций скорость снизится, так как в 4-х байтную переменную удастся упаковать меньше дискретизированных

элементов (например, $q = 16$ и $p = 4$). Восьмикратное ускорение не достигается в связи с тем, что при работе с упакованными данными периодически приходится их в переменные большего формата.

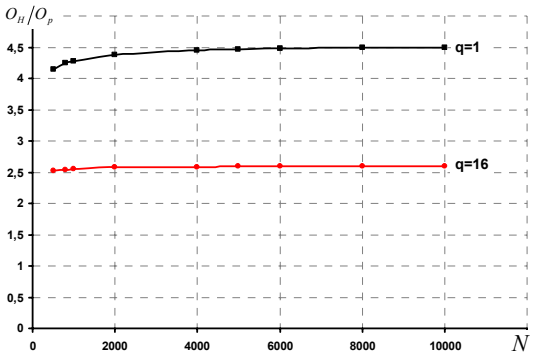

Рис.12. Отношение числа операций стандартной модели Хопфилда к алгоритму с упаковкой в 4 байта для разного числа градаций.

Чтобы получить полное ускорение алгоритма необходимо также учесть время, затрачиваемое на передачу данных между памятью и процессором. Так как при упаковке передаются сразу несколько чисел, то число обращений процессора к памяти будет примерно в p раз меньше, чем при обычном алгоритме. Общее время работы алгоритмов можно оценить как

$$\begin{aligned} t_H &= O_H \cdot t_{\text{ар}} + t_{\text{пер}} \cdot O_H \\ t_p &= O_p \cdot t_{\text{ар}} + t_{\text{пер}} \cdot O_H / p \end{aligned}. \qquad (22)$$

Здесь $t_{\text{ар}}$ - время выполнения арифметической операции, $t_{\text{пер}}$ - время передачи данных между процессором и памятью. Время передачи данных обычно много больше времени выполнения арифметических операций. Можно рассмотреть 2 граничных случая.

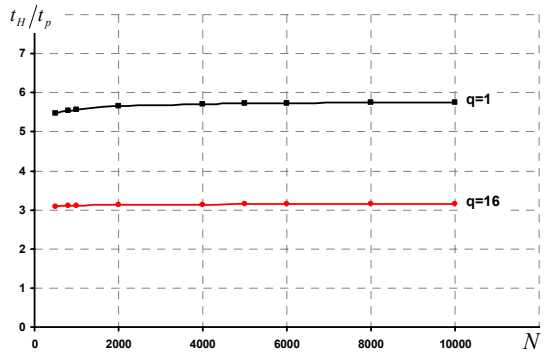

Рис.13. Полное ускорение алгоритма с упаковкой по сравнению с сетью Хопфилда, когда время выполнения арифметических операций равно времени передачи данных между процессором и памятью.

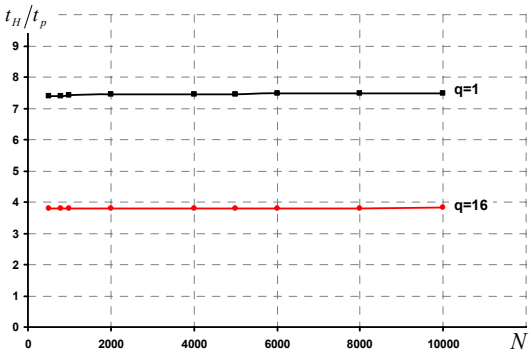

Рис.14. Полное ускорение алгоритма с упаковкой по сравнению с сетью Хопфилда, когда время выполнения арифметических операций в 10 раз меньше времени передачи данных между процессором и памятью.

1. $t_{\text{пер}} = t_{\text{ар}}$ (см. рис. 13): в этом случае, если вычислить отношение времен работы алгоритмов получим, что при $q = 1$ алгоритм с упаковкой данных работает приблизительно в 5.5 раз быстрее, чем сеть Хопфилда. При $q = 16$ ускорение составит более 3 раз.

2. $t_{пер} >> t_{ар}$ (см. рис. 14): например, $t_{пер} = 10 t_{ар}$. В этом случае получим, что при $q = 1$ алгоритм с упаковкой данных работает почти в 7.5 раз быстрее, чем сеть Хопфилда. А при $q = 16$ ускорение приближается к 4 раз.

Таким образом, ускорение алгоритма с упаковкой данных в зависимости от архитектуры вычислительной машины позволяет достичь ускорения от 5.5 до 7.5 раз.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

В диссертационной работе получены следующие результаты.

1. Предложена и исследована процедура дискретизации матрицы связей, которая позволяет ускорить процесс минимизации квадратичного функционала. Удалось выделить два настроечных параметра: число градаций и размер нулевого отрезка, которые позволяют определить качество дискретизации и эффективность ее использования еще до начала проведения каких-либо экспериментов.
2. Определены оптимальные параметры дискретизации, при которых ошибка в направлениях локальных полей составляет менее 10% в любой точке конфигурационного пространства. Показано, что расстояние по Хеммингу и по энергии между минимумом дискретизированного функционала и минимумом исходного функционала, в который пришла бы исходная сеть из дискретизированного минимума, прямо пропорционально вероятности несовпадения направлений локальных полей в случайной точке. Хеммингово расстояние менее 12%, а по энергии меньше 7% для любого числа градаций.
3. На основе процедуры дискретизации разработаны алгоритмы минимизации квадратичного функционала: одноэтапный и двухэтапный. Исследовано быстродействие предложенных алгоритмов, а также их эффективность по сравнению со стандартной моделью Хопфилда.
4. Применение одноэтапного алгоритма при использовании упакованных чисел позволяет достичь 7-кратного увеличения скорости. При этом глубина полученных минимумов по энергии на 7% меньше, чем при стандартной модели Хопфилда. При этом требуется в 8 раз меньше оперативной памяти, чем при стандартной модели Хопфилда. Таким образом, с помощью одноэтапного алгоритма можно решать задачи недоступные стандартным алгоритмам.
5. Для отыскания более глубоких минимумов предлагается двухэтапный алгоритм. Он позволяет отыскивать те же минимумы, что и стандартная модель, но с вероятностью более 50% находятся более глубокие. Использование только самых глубоких дискретизированных минимумов на втором этапе позволило сохранить скорость алгоритма, достигаемую за счет дискретизации.

РАБОТЫ АВТОРА ПО ТЕМЕ ДИССЕРТАЦИИ

1. Kryzhanovsky M.V., Malsagov M.U. Clipping procedure in optimization problems and its generalization // Optical Memory and Neural Networks (Information Optics).– 2009.– Vol. 18, №3.– pp. 181-187.
2. Крыжановский Б.В., Крыжановский М.В., Мальсагов М.Ю. Дискретизация матрицы в задаче бинарной минимизации квадратичного функционала // Доклады Академии Наук.– 2011.– Т.438, №3.– сс. 312-317.
3. Мальсагов М.Ю. Понижение разрядности элементов матрицы для ускорения алгоритма дискретной оптимизации // Нейрокомпьютеры: разработка и применение.– 2011.– №4.– сс. 22-31.
4. Kryzhanovsky M.V., Malsagov M.Yu. Modification of Binary Optimization Algorithm and Use Small Digit Capacity Numbers // Optical Memory and Neural Networks (Information Optics).– 2011.– Vol. 20, №3.– pp. 194–200.
5. Крыжановский М.В., Мальсагов М.Ю. Применение чисел малой разрядности в задаче бинарной оптимизации // Программные продукты и системы.– 2011.– №4.– сс. 40-44.
6. Крыжановский М.В., Мальсагов М.Ю. Обобщение процедуры клиппирования в задачах оптимизации в дискретном пространстве // Искусственный интеллект.– 2009.– №3.– сс. 496-503.
7. Крыжановский М.В., Мальсагов М.Ю. Особенности применения дискретизации в задачах поиска глобального минимума // Искусственный интеллект.– 2011.– №3.– сс. 497-505.
8. Крыжановский М.В., Мальсагов М.Ю. Применение процедуры клиппирования и ее обобщение в задачах поиска глобального минимума // Сборник трудов XI Всероссийской научно-технической конференции «Нейроинформатика-2009».– М.: МИФИ, 2009.– ч.2.– сс. 61-68.
9. Крыжановский М.В., Мальсагов М.Ю. Ускорение модифицированной процедуры клиппирования // Сборник трудов XII Всероссийской научно-технической конференции «Нейроинформатика-2010».– М.: МИФИ, 2010.– ч.2.– сс. 45-54.
10. Kryzhanovsky M.V., Malsagov M.U. Small digit capacity arithmetic for problems of discrete optimization // Proc. of the IV International Conference on Neural Networks and Artificial Intelligence «ICNNAI-2010».– Brest.– Belarus.– pp. 101-106.
11. Крыжановский М.В., Мальсагов М.Ю. Дискретизация матрицы в задаче бинарной оптимизации // Сборник трудов XIII Всероссийской научно-технической конференции «Нейроинформатика-2011».– М.: МИФИ, 2011.– ч.3.– сс. 152-160.