Федеральное государственное бюджетное учреждение науки Научно-Исследовательский Институт Системных Исследований РАН

На правах рукописи

Карандашев Яков Михайлович

# МЕТОД ТРАНСФОРМАЦИИ МАТРИЦЫ В ЗАДАЧЕ КВАДРАТИЧНОЙ БИНАРНОЙ ОПТИМИЗАЦИИ

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

Диссертация на соискание ученой степени кандидата физико-математических наук

> Научный руководитель: член-корреспондент РАН доктор физико-математических наук Крыжановский Борис Владимирович

# СОДЕРЖАНИЕ

СОДЕРЖАНИЕ	2
1. ВВЕДЕНИЕ	3
1.1 Постановка задачи	3
1.2 Обзор литературы	3
1.3 Актуальность темы	11
1.4 Цели и залачи лиссертации	
1.5 Основные положения выносимые на зашиту	12
1.6 Метолы исслелования	14
1.7 Апробация работы и публикации	15
1.8 Структура диссертации	19
2. ОБЛАСТЬ ПРИТЯЖЕНИЯ МИНИМУМА	21
2.1 Область притяжения минимума	21
2.2 Форма минимума	26
2.3 Ограничение на глубину минимума	27
2.4 Выводы по главе 2	
2.5 Приложение А. Форма энергетической поверхности в n-окрестности	29
3. АЛГОРИТМ С ВОЗВЕДЕНИЕМ МАТРИЦЫ В СТЕПЕНЬ (DDK)	32
3.1 Стандартный алгоритм случайного поиска (SRS)	32
3.2 Базовые соотношения	35
3.3 Трансформация поверхности	36
3.4 Обоснование алгоритма DDK	
3.4а) Углубление минимумов	38
3.4б) Сдвиг минимума	
3.46) Выводы и их проверка	40
3.5 Эффективность АЛГОРИТМА DDK	43
3.5а) Результаты для матриц двумерной модели Эдвардса-Андерсона (2D EA)	43
3.5б) Результаты для матриц модели Шеррингтона-Киркпатрика (SK)	46
3.5в) Улучшение динамики. Алгоритм Хоудайера-Мартина.	48
3.5г) Результаты для матриц 3-мерной модели Эдвардса-Андерсона (3D EA)	51
3.6 Применения алгоритма к задаче разбиения графа (graph bipartitioning)	52
3.7 Выводы ПО ГЛАВЕ 3	59
4. АЛГОРИТМ MIX-MATRIX (MM)	62
4.1 Углубление минимума	63
4.2 Сдвиг минимума	65
4.3 Описание алгоритма MM	66
4.4 Результаты для алгоритма MM	66
4.5 Сравнение с алгоритмами для задачи MAXCUT	73
4.6 Выводы по главе 4	74
4./ Приложение А	//
5. ОГГАНИЧЕНИЛ НА ВЕСА ПАТТЕГНОВ В КВАЗИЛЕВВОВСКОЙ МАТГИЦЕ ІОЛНИТЕЛЬНАЯ ГЛАВА)	78
5.1 Квазихеббовская модель	78
5.2 Основное уравнение	79
5.3 Простейший случай.	82
5.4 Произвольное распределение весов	90
5.4a) Веса в виде гармонической последовательности	93
5.4б) Веса в виде геометрической прогрессии	
5.4в) Веса в виде арифметической прогрессии	104
5.5 Выводы по главе 5	108
6. ЗАКЛЮЧЕНИЕ	111
СПИСОК ЛИТЕРАТУРЫ	117

### 1. ВВЕДЕНИЕ

Дискретная оптимизация играет важную роль как в инженерных задачах, так и в фундаментальной науке. Открытой проблемой является природа энергетической поверхности в оптимизационных задачах. Обычно соглашаются с тем, что эта поверхность сильно испещрена минимумами. Нахождение основного состояния в таких беспорядочных системах почти всегда является NP-сложной задачей, поэтому эффективные алгоритмы особенно необходимы [1](Houdayer & Martin (1999)).

#### 1.1 Постановка задачи

Настоящая работа направлена на повышение эффективности процедуры случайного поиска, применяемой при решении задач бинарной минимизации. Это класс задач, решение которых сводится к минимизации квадратичного функционала E(S):

$$E(S) = -\frac{1}{N^2 \sigma_T} \sum_{i=1}^{N} \sum_{j=1}^{N} T_{ij} s_i s_j , \qquad (1.1)$$

построенного на заданной  $N \times N$ -матрице T в N-мерном конфигурационном пространстве состояний  $S = (s_1, s_2, ..., s_N)$  с дискретными переменными  $s_i = \pm 1$ ,  $i = \overline{1, N}$ ;  $\sigma_T$  - стандартное отклонение матричных элементов  $T_{ij}$ . Матрицу T без ограничения общности будем полагать симметричной  $(T_{ij} = T_{ji})$  с нулевой диагональю  $(T_{ii} = 0)$ . Нормировка на  $\sigma_T$ обеспечивает корректность сравнения результатов для различных матриц.

#### 1.2 Обзор литературы

Функционал (1.1)определяет вида энергию системы взаимодействующих спинов в магнитной решётке. В случае случайных положительных и отрицательных взаимодействий  $T_{ij}$  такая система в физике стеклом называется спиновым И характеризуется беспорядком И фрустрацией, приводящей к очень сложной энергетической поверхности [25]. Эти физические модели также формулируются как задачи оптимизации, которые применимы во многих областях научной деятельности [3,5]. К сожалению, проблема поиска основного состояния является NP-сложной [6]. Поэтому разнообразные алгоритмы адаптировались на протяжении многих лет специально для поиска основных состояний и для исследования спиновых стёкол при конечных температурах и при переходе к пределу нулевой температуры [7,8]. Ввиду большой сложности задач разрабатывались специализированные компьютеры для их решения [9].

Задача минимизации в конфигурационном пространстве дискретных переменных кардинально отличается от той же задачи в непрерывном пространстве с вещественными переменными, когда она может быть решена с применением известных вычислительных процедур. Во-первых, в случае квадратичный функционал бинарных переменных является многоэкстремальным, даже если матрица Т является положительно-При определенной. ЭТОМ число локальных МИНИМУМОВ растет экспоненциально с ростом размерности пространства. Во-вторых, задача бинарной минимизации является NP-сложной, т.е. в общем случае нет полиномиального алгоритма её решения.

В случае когда система является ферромагнетиком, т.е. взаимодействия  $T_{ij}$  имеют все одинаковый положительный знак, задача отыскания основного состояния является тривиальной, поскольку решением является состояние, в котором все спины сонаправлены (все +1 или все -1). Даже в случае добавления к функционалу (1.1) линейного члена (внешнее случайное поле), задача для ферромагнетика остаётся полиномиально разрешимой при помощи алгоритмов, основанных на методах максимального потока в сети [10,11].

Обычно система становится фрустрирована если существует замкнутый цикл, в котором нечётное число отрицательных связей. В этом случае, спиновое стекло не может быть преобразовано в ферромагнетик [11]. По этой причине быстрые алгоритмы поиска основного состояния не могут быть применены, а проблема становится NP-сложной, т.е. для её решения известны только алгоритмы с экспоненциальным временем работы (более подробно о проблеме  $P \stackrel{?}{=} NP$ , о классе NP-сложных задач, и о полиномиальной сводимости задач друг к другу см. [12-14]).

Как исключение, в случае двумерного спинового стекла без внешнего поля и с периодическим граничным условием не более чем в одном измерении существуют эффективные полиномиальные алгоритмы отыскания основного состояния [15], многие из которых основаны на паросочетаниях в графах [16-18]. Полиномиально разрешима также двумерная решётка с периодическими граничными условиями в обоих направлениях, если связи имеют вид  $\pm J$  [19]. Для гауссовых связей вопрос до сих пор открыт. Как только мы добавляем в систему внешнее поле, т.е. линейный член в (1.1), задача становится NP-сложной для всех типов взаимодействий. Для трёхмерной кубической решётки задача является NP-сложной даже в отсутствии внешнего поля.

Известно, что задача квадратичной бинарной оптимизации напрямую связана с задачей разбиения графа. В своей базовой версии, эта задача состоит в том, чтобы разбить вершины графа на две непересекающиеся части так, чтобы вес рёбер соединяющих этих части был минимален или максимален. В первом случае задача обозначается как MIN-CUT, а во втором MAX-CUT. Задачи о разрезах имеют множество применений, например в разметке электронных схем, в повышении надёжности сетей, распознавании зрительных сцен и др. Множество важных комбинаторных оптимизационных задач может быть естественным образом сформулировано как задачи квадратичной бинарной оптимизации с ограничениями, причём, применение знаний, полученных из исследования случая без ограничений, часто позволяет существенно повысить скорость конечных алгоритмов [20].

Для неотрицательных весов рёбер, задача MIN-CUT может быть решена при помощи метода потоков в сети благодаря дуальности максимального потока и минимального разреза в сети (алгоритм Форда-Фалкерсона [10]), либо при помощи алгоритма Штора-Вагнера [21].

Для общего случая весов рёбер, задача МАХ-СUТ (как и задача МІN-CUT, получающаяся инверсией знаков весов рёбер) является NP-сложной. Для некоторых классов графов, например, как сказано выше, для планарных графов [15], для графов не содержащих  $K_5$  (полный граф из пяти вершин)[23], для слабо двудольных графов, в случае целочисленных весов когда род графа ограничен константой и значения весов рёбер ограничены по абсолютному значению полиномом от размера графа [22], а также для некоторых других случаев существуют полиномиальные алгоритмы решения этой задачи. Детальные исследования различных случаев граничных условий и весов рёбер, делающих задачу легко решаемой или NP-сложной, приведены в [24-25].

Строгие алгоритмы. Для решения задачи в общем случае предложено несколько алгоритмов (обзор дан в [26-27]). Простейший метод состоит в перенумерации всех  $2^N$  возможных состояний и очевидно имеет экспоненциальное время работы. Даже кубическая решётка состоящая из  $4^3$  элементов уже оказывается слишком большой. Базовая идея метода ветвей и границ [28] состоит в том, чтобы исключить из просмотра те области пространства состояний, в которых состояния с низкой энергией не могут быть найдены, и таким образом чтобы все низкоэнергетические состояния системы размера  $4^3$  уже можно было посчитать [29]. Также к спиновым стёклам размера  $4^3$  применялся метод трансфер-матриц [30].

Для решения NP-сложных задач предложены алгоритмы  $\varepsilon$ аппроксимации, т.е. полиномиальные алгоритмы, строго дающие решение, отличающееся не более чем на  $\varepsilon$  от энергии глобального минимума. Выдающимся результатом здесь является работа [31] (Goemans and Williamson (1994)), в которой был предложен алгоритм, который за полиномиальное время даёт решение задачи MAX-CUT со значением, составляющим не менее 0.878 от величины максимального разреза, т.е. отличающимся менее чем на 13% от оптимума. С другой стороны, как следует из работы [32] (Bellare et al (1995)), в предположении  $P \neq NP$ , не существует полиномиального алгоритма, который доказуемо даёт решение со значением разреза, составляющим хотя бы 98% от оптимального.

Исследование алгоритмов Е-аппроксимации привело к открытию бинарной методов сведения задачи оптимизации к линейному И полуопределённому программированию (так называемые LP- и SDPрелаксации). Такой подход активно используется для поиска верхних и нижних границ в методе ветвей и границ и позволяет существенно сузить полный перебор и найти точное решение, пусть не за полиномиальное, но за умеренное время, для более широкого круга задач. В работе [33](Liers et al (2004))алгоритм, основанный LP-релаксации, описан на который переписывает квадратичную функцию энергии в виде линейной функции с дополнительными набором неравенств, которые должны быть выполнены для всех допустимых решений. Поскольку не все неравенства известны заранее, метод итерационно решает линейную задачу, ищет неравенства, которые нарушены, и добавляет их к набору до тех пор пока решение не будет найдено. Поскольку число неравенств растёт экспоненциально с размером задачи, то же наблюдается и со временем работы алгоритма. Однако данный метод за несколько минут рабочего времени обычного компьютера позволяет находить основное состояние для задач 2D и 3D моделей Эдвардса-Андерсона [34] с периодическими граничными условиями и гауссовыми связями для размерности задачи вплоть до нескольких тысяч спинов. В работах [35-36] (Rendl et al (2010), Wiegele (2010)) описаны методы SDP-релаксации задачи MAX-CUT, позволяющие решать задачи с более плотными матрицами связей, чем у 2D и 3D решёток. Однако размер задач, с которыми данный алгоритм справляется, существенно уменьшается с увеличением плотности ненулевых матричных элементов.

Эвристические методы, локальный поиск. Для решения задач больших размерностей, а также с более плотными матрицами (например, модель

твёрдого тела Шеррингтона-Киркпатрика [37]) применяются эвристические методы.

Широко распространённым в физике подходом является эволюция системы при некоторой температуре используя метод Монте-Карло (МС) [38]. Для динамики Метрополиса обычно просто выбирают на каждом шаге случайный спин и переворачивают его с вероятностью min(1,exp( $-\beta\Delta E$ )), где  $\beta$  - обратная температура,  $\Delta E$  - изменение энергии (1.1) к которому приведёт переворот данного спина. Теоретически можно достичь равновесия при низкой температуре, следовательно основное состояние будет найдено. На этом основаны методы симуляции отжига [39-42]. Тем не менее, для таких систем достижение равновесия в действительности занимает очень долгое время, поскольку динамика застревает в метастабильных состояниях из-за существования многих низкоэнергетических локальных минимумов. Для борьбы с этим можно использовать алгоритм MC<sup>3</sup> (Metropolis-coupled Markov-chain Monte-Carlo) [43], также известный в физике как репличный обмен или параллельная закалка (Parallel Tempering) [44].

Предлагались подходы основанные на нейронных сетях [45]. С разным успехом применялись варианты генетических алгоритмов [46-49]. В работе [1] была предложена комбинация генетического алгоритма с рекурсивной процедурой ренормализации. Основная идея состояла в том, чтобы делить задачу на подзадачи меньшего размера и решать их тем же путём. Данная техника применялась к конечномерным системам с гауссовым распределением весов, с размерами вплоть до 10<sup>3</sup>, а также к другим комбинаторным задачам типа коммивояжёра [50].

Самыми простыми и потому наиболее популярными эвристиками являются локальные одношаговые алгоритмы поиска. В настоящей работе в качестве основной процедуры локальной минимизации выбрана асинхронная динамика модели Хопфилда [51-52] (Hopfield (1982)), которая является ядром большинства известных алгоритмов бинарной минимизации. Модель описывает систему из *N* спинов, взаимодействие которых задается

функционалом энергии E(S). Стандартная (асинхронная) динамика минимизации состоит в следующем: вычисляется величина локального поля  $h_i \sim -\partial E(S) / \partial s_i$ , действующего на произвольно выбранный *i*-й спин:

$$h_i = \sum_{j \neq i}^N T_{ij} s_j \tag{1.2}$$

Если спин направлен вдоль поля  $(h_i s_i \ge 0)$ , то его состояние не изменяется. Если спин находится в неустойчивом состоянии ( $h_i s_i < 0$ ), то он разворачивается вдоль поля, принимая состояние  $s_i = \text{sign } h_i$ . Эта процедура последовательно применяется ко всем спинам до тех пор, пока система не конвергирует в устойчивое состояние. Подчеркнем, что в каждый момент времени обновляется состояние только одного спина. В этом отличие асинхронной динамики от синхронной, в которой состояния всех спинов обновляются одновременно. Описываемая динамика есть не что иное, как спуск по энергетической поверхности E(S), являющийся полным аналогом покоординатного градиентного спуска в непрерывном пространстве. Уменьшение энергии при каждом перевороте неустойчивого спина гарантирует нам, что процесс, описываемый асинхронной динамикой, за конечное число шагов приводит систему в устойчивое положение с минимумом энергии. Конечно, найденный минимум скорее всего окажется локальным, в то время как нам бы хотелось найти самый глубокий (глобальный) минимум функционала. Стандартным приёмом здесь является случайный поиск (random search), суть которого состоит в многократном повторении описанных выше спусков из различных случайных стартовых конфигураций.

Несмотря на простоту алгоритма, спуск из одной случайной конфигурации требует приблизительно  $O(N^2)$  операций для полносвязной матрицы (как в модели Шеррингтона-Киркпатрика) и O(N) для разреженных матриц (как в моделях Эдвардса-Андерсона двумерной и трёхмерной решётки). Количество локальных минимумов с ростом N экспоненциально

экспоненциально уменьшается увеличивается, а значит, вероятность обнаружения глобального минимума. Несмотря на указанные трудности, эвристические методы широко применяются для решения задач бинарной Успешное применение модели Хопфилда оптимизации. к задаче коммивояжера [52] (Hopfield and Tank (1985)), инициировало широкие исследования аналогичных подходов к решению задач теории графов [53] (Fu and Anderson (1986)), нейросетевой оптимизации обработки изображений [54] (Poggio and Girosi (1990)) и ряду других приложений [55-66]. Анализ причин подобного успеха проведен в работе [58] (Kryzhanovskii et al (2005)), где показано, что при случайном поиске вероятность отыскания какого-либо минимума экспоненциально возрастает с ростом глубины этого минимума. В случайной инициализации сети она с наибольшей частности, при вероятностью конвергирует в состояние, соответствующее глобальному минимуму [67-68] (Kryzhanovsky et al (2006); Kryzhanovsky et al (2009)). С несколько меньшей вероятностью она конвергирует в один из локальных минимумов, сравнимых ПО глубине с глобальным. Α вероятность неглубокие конвергенции относительно В локальные минимумы экспоненциально мала. Такое соотношение вероятностей означает, что система с подавляющей вероятностью находит если не оптимальное решение (глобальный минимум), то одно из субоптимальных решений (локальный минимум). Данный вопрос мы исследуем в главе 2.

Хопфилда Пользуясь сетевой динамикой В качестве основной процедуры локальной минимизации, мы предлагаем подход, основанный не на изменении динамики локального поиска, а на трансформации самой энергетической поверхности, т.е. изменении статики самой задачи. Разработке и исследованию такого подхода, использующего простой способ трансформации энергетической поверхности путём возведения матрицы в степень, посвящена данная диссертации. Подходы, основанные на тех же базовых идеях, но использующие другие трансформации, также уже предлагались и были использованы для различных оптимизационных задач [69-71].

#### 1.3 Актуальность темы

Несмотря на многочисленные работы по данной тематике, NP-полнота задачи квадратичной оптимизации с бинарными переменными оставляет эту задачу трудно решаемой. Линейное увеличение скорости вычислений компьютеров (например, при помощи графических ускорителей) не может превозмочь экспоненциальную от размера входных данных сложность задачи. Поэтому задача остаётся актуальной (и вероятно будет оставаться такой вплоть до создания квантовых вычислителей).

#### 1.4 Цели и задачи диссертации

Обычно эффективность процедуры случайного поиска пытаются повысить видоизменяя динамику спуска по поверхности, описываемой функционалом E(S) [11, 60-61] (Hartmann and Rieger (2001), Hartmann and Rieger (2004), Duch and Korczak (1998)). В отличие от этого, в настоящей работе предлагается кардинально иной подход: изменять не динамику спуска поверхности, а трансформировать саму поверхность так, чтобы ПО увеличился радиус области притяжения глобального минимума (равно как и других минимумов, сравнимых с глобальным по глубине). Как следствие, вероятность отыскания глобального минимума возрастает как  $exp(\alpha N)$ , где N – размерность задачи; спектр энергий находимых минимумов сильно сдвигается в глубокую сторону так, что среднее по спектру значение отличается от энергии глобального минимума всего лишь на 4% - 6% (в зависимости от типа матриц).

Диссертационная работа направлена на исследование способов трансформации энергетической поверхности функционала с целью резкого увеличения эффективности алгоритмов минимизации.

Для достижения данной цели в работе ставятся следующие задачи:

Исследовать энергетическую поверхность функционала, в частности форму локальных минимумов.

Исследовать зависимость вероятности отыскания минимума при случайном поиске от глубины минимума.

Исследовать влияние трансформации матрицы функционала на изменение глубины минимумов, сдвиг минимумов в пространстве и смещение спектра находимых минимумов.

На основе проведенных исследований построить эффективный алгоритм минимизации квадратичного функционала.

#### 1.5 Основные положения выносимые на защиту

1) В работе исследуется NP-сложная задача квадратичной бинарной оптимизации (эквивалент задачи 0/1-программирования и задачи о максимальном разрезе графа). Исследованы свойства энергетической поверхности, описываемой квадратичным функционалом в *N*-мерном пространстве состояний с бинарными переменными. Получены выражения для формы минимумов. При случайной матрице связей средняя энергия в *n*-окрестности минимума описывается формулой  $\overline{E}(n) \approx E_0 (1 - 2n / N)^2$ , где  $E_0$  - энергия минимума, при этом разброс относительно среднего стремится к нулю как 1/N. Таким образом, форма минимума определяется лишь глубиной минимума.

2) Показано, что радиус области притяжения минимума  $x_0$  полностью определяется глубиной минимума  $\gamma = |E|\sqrt{N}$  из уравнения:

$$\frac{x_0}{1-x_0} = \Phi[\gamma(1-2x_0)],$$

где  $\Phi(\cdot)$  - интеграл вероятностей. Поскольку спектр энергий минимумов имеет узкое распределение и глубина минимумов отличается от среднего значения глубины  $\overline{\gamma}$  не более чем на 10-20%, то можно считать зависимость радиуса от глубины линейной:

$$x_0 \approx \overline{x} + k \,\overline{x} \, (\gamma - \overline{\gamma}),$$

где  $\overline{x}$  и k - константы (порядка  $\overline{x} \approx 0.384$  и k  $\approx 0.076$ ). Из этого следует, что отыскания случайном вероятность минимума при поиске, равная области относительному объёму притяжения минимума всём BO пространстве конфигураций, экспоненциально зависит от глубины минимума  $P(\gamma) = x^N / 2^N \approx \overline{P} e^{kN(\gamma - \overline{\gamma})}.$ 

3) Показано, что возведение матрицы *T* в степень *k* (*k* = 2, 3, 4....) меняет энергетическую поверхность функционала, однако наиболее глубокие минимумы исходного функционала *E* остаются минимумами в новом функционале  $E_k$ , либо присутствуют минимумы в их малой *d*-окрестности (при *k* = 3 сдвиг *d* ≈ 0.026*N*), но при этом оказываются глубже (для матриц 2D EA при *k* = 3 происходит углубление минимума, соответствующего глобальному на исходном функционале, примерно на 15%).

4) При сложении матрицы T и матрицы  $T^2$  с весами (1-z) и z, соответственно, получается новый функционал  $E_z$ , который обладает теми же свойствами, что и  $E_k$ , только в этом случае можно контролировать сдвиг и углубление минимумов, варьируя параметр z от 0 до 1. Получены выражения для величины углубления минимума и его сдвига в пространстве. Показано, что существует некоторое оптимальное значение  $z_{opt} \approx 0.51$  (для матриц 2D EA), при котором происходит максимальное углубление (порядка 45%), при этом сдвиг оказывается около  $\overline{d}_{opt} = 0.01N$ .

5) На основании двух предложенных выше трансформаций функционала, в работе предложены два алгоритма минимизации (DDK и MM), состоящие из двух шагов. На первом шаге происходит минимизация трансформированного функционала (спуск по поверхности  $E_k$  в алгоритме DDK и спуск по поверхности  $E_z$  в алгоритме MM). Далее из найденной точки процедура минимизации продолжается на исходном функционале (продолжение спуска по поверхности E). Такая двухэтапная процедура

алгоритмов позволяет находить минимумы исходного функционала, нивелируя сдвиги, введённые трансформацией поверхности.

6) Показано, что использование предложенных алгоритмов (DDK и MM) приводит к существенному сдвигу спектра находимых минимумов в сторону глобального минимума и к экспоненциальному по *N* росту эффективности алгоритма случайного поиска, при этом количество операций, затрачиваемых на увеличивается вдвое. Дано строгое обоснование ОДИН старт, эффективности предложенных алгоритмов для полносвязных матриц модели Шеррингтона-Киркпатрика; применение алгоритмов для другого типа матриц является эвристическим. Эффективность предложенных двухэтапных нескольких алгоритмов проверена на типах случайных матриц И продемонстрирована сравнением с известными алгоритмами на задачах MaxCut и Graph Bipartitioning.

#### 1.6 Методы исследования

Для целей исследования были выбраны матрицы SK-модели – модели Шеррингтона-Киркпатрика, в которой матричные элементы можно рассматривать как независимые случайные величины с нулевым средним. Для таких типов матриц удалось сделать некоторые теоретические оценки, пользуясь центральной предельной теоремой.

Также разработанные алгоритмы проверялись на разреженных матрицах моделей Эдвардса-Андерсона (2D EA и 3D EA модели, описывающие двумерные и трехмерные спиновые решетки с периодическими граничными условиями, в которых взаимодействие отлично от нуля только между ближайшими соседями в решётке и описывается нормально распределенными матричными элементами).

Анализ энергетической поверхности и формы минимумов производился при помощи разложения матрицы T на две части: той, которая полагается ответственной за существование минимума ( $T_0$ ), и оставшейся части ( $T_1$ ). При этом матрицы  $T_0$  и  $T_1$  оказываются некоррелированными (подробнее см.

главу 2). Исходя из некоррелированности этих матриц *делалось допущение об* их независимости, и на основании этого матрица  $T_1$  полагалась чисто случайной по отношению к минимуму. Данное допущение позволило сделать важные оценки, которые во многом соответствовали эксперименту. В том числе вероятность сдвига конфигурации вычислялась путём поиска отношения «сигнал/шум», где сигнальной части соответствовала матрица  $T_0$ , а шумовой – матрица  $T_1$ .

При решении поставленных задач в данной работе были использованы методы системного анализа, вычислительной математики, комбинаторики, математической статистики и теории вероятностей. Вычислительные эксперименты проводились на персональном компьютере (Intel Core i3 CPU 3.08ГГц, 2 Гб ОЗУ), и обычно занимали от нескольких минут до нескольких дней счёта (в зависимости от размерности задачи, числа стартов при случайном поиске и количества матриц, на которых набиралась статистика).

#### 1.7 Апробация работы и публикации

По материалам диссертации опубликовано 22 работы, из них 10 – в российских и зарубежных журналах (в том числе 5 из перечня ВАК), 12 – в трудах конференций.

Список публикаций:

- Я.М. Карандашев, Б.В. Крыжановский. Увеличение глубины глобального минимума квадратичного функционала путём возведения матрицы в степень. // XI Всероссийская научно-техническая конференция Нейроинформатика-2009. Москва, МИФИ (2009)
- J.M. Karandashev, B.V. Kryzhanovsky. An Effective Increase of the Basin of a Quadratic Functional Global Minimum. // ICONS 2009: The 2nd IFAC International Conference on Intelligent Control Systems and Signal Processing. Turkey, Istanbul (2009)
- 3. Карандашев Я.М., Крыжановский Б.В. Эффективное увеличение области притяжения глобального минимума квадратичного бинарного

функционала при нейросетевом поиске. // Международная конференция Актуальные проблемы информационно-компьютерных технологий мехатроники и робототехники (ИКТМР-2009). Дивноморск. (2009)

- 4. Я.М. Карандашев, Б.В. Крыжановский. Эффективное увеличение области притяжения глобального минимума квадратичного бинарного функционала. // Всероссийская конференция Методы и средства обработки информации (МСО - 2009). Москва, МГУ (2009)
- 5. Я.М. Карандашев, Б.В. Крыжановский. Эффективное увеличение области притяжения глобального минимума квадратичного бинарного функционала при нейросетевом поиске. //14-ая Всероссийская конференция Математические Методы Распознавания Образов (ММРО-14), Суздаль (2009)
- 6. Я.М. Карандашев, Б.В. Крыжановский. Улучшение нейросетевой оптимизации бинарного квадратичного функционала. // 52-я научная конференция МФТИ. Долгопрудный, МФТИ (2009)
- Я.М. Карандашев, Б.В. Крыжановский. О трансформации энергетической поверхности в задаче бинарной оптимизации. //ДАН. Т. 429, №4, с. 465-469 (2009)
- Ya. M. Karandashev and B. V. Kryzhanovsky. Transformation of Energy Landscape in the Problem of Binary Minimization. // Doklady Mathematics, 2009, Vol. 80, No. 3, pp. 927–931 (2009)
- 9. Я.М. Карандашев, Б.В. Крыжановский. Эффективное увеличение области притяжения глобального минимума квадратичного бинарного функционала при нейросетевом поиске. // Искусственный интеллект, № 4, с. 37-44 (2009)
- 10. Я.М. Карандашев, Б.В. Крыжановский. Повышение эффективности Кернигана-Лина путём алгоритма модификации // матрицы межсвязей. Всероссийская конференция Нейроинформатика-2010. Москва, МИФИ (2010)

- Ya. M. Karandashev, B. V. Kryzhanovsky. Binary Optimization: Efficient Increasing of Global Minimum Basin of Attraction. // Optical Memory & Neural Networks, Vol. 19, No. 2, pp. 110–125 (2010)
- Ya. M. Karandashev and B. V. Kryzhanovsky. Efficient Energy Landscape Transformation in the Problem of Binary Minimization. // The 2010 International Joint Conference on Neural Networks (IJCNN), Spain, Barcelona, pp. 1-6 (2010)
- Yakov Karandashev, Boris Kryzhanovsky. Binary Minimization: Increasing the Attraction Area of the Global Minimum in the Binary Optimization Problem. // ICANN (2) 2010, LNCS, v. 6353/2010, pp. 525-530 (2010).
- Iakov M. Karandashev, Boris V. Kryzhanovsky. Transformation of Edge Weights in a Graph Bipartitioning Problem. // Proceedings of the 21st international conference on Artificial neural networks ICANN (Part 2), LNCS, v. 6792/2011, pp. 25-31 (2011).
- 15. Iakov Karandashev, Boris Kryzhanovsky. Increasing the attraction area of the global minimum in the binary optimization problem. // arXiv:1109.0165v1
- 16. Я.М. Карандашев, Б.В. Крыжановский, Л.Б. Литинский. Сильная неустойчивость спектра минимумов бинарного квадратичного функционала. ДАН. - Т. 436, N 6. С. 733-737 (2011)
- Ya. M. Karandashev, B. V. Kryzhanovsky and L. B. Litinskii. Strong instability of the minima spectrum of a quadratic binary functional. // Doklady Mathematics Vol. 83, 1, 116-120 (2011)
- Iakov Karandashev, Boris Kryzhanovsky and Leonid Litinskii. Weighted Patterns as a Tool to Improve the Hopfield Model. // Physical Review E 85, 041925 (2012).
- Я.М. Карандашев. Трансформация матрицы связей в задаче разбиения графа.// Вестник компьютерных и информационных технологий. №1, 33-37 (2012)

- 20. I. Karandashev and B. Kryzhanovsky . Mix-matrix Method in Problem of Discrete Optimization. // ICCGI 2012 : The Seventh International Multi-Conference on Computing in the Global Information Technology, Italy, Venice (2012)
- Karandashev and B. Kryzhanovsky. The Mix-Matrix Method in the Problem of Binary Quadratic Optimization. // Proc. of ICANN 2012, Part I, LNCS 7552, pp. 41–48 (2012)
- Karandashev and B. Kryzhanovsky. Increasing the attraction area of the global minimum in the binary optimization problem. // J. Glob. Optim., V. 56, 3 , pp. 1167-1185 (2013), DOI: 10.1007/s10898-012-9947-7

Основные результаты работы докладывались на следующих *конференциях*:

- 1. XI
   Всероссийская
   научно-техническая
   конференция

   Нейроинформатика 2009.
   Москва,
   МИФИ (2009)
- 2. **ICONS 2009:** The 2nd IFAC International Conference on Intelligent Control Systems and Signal Processing. Turkey, Istanbul (2009)
- 3. ИКТМР-2009: Международная конференция Актуальные проблемы информационно-компьютерных технологий мехатроники и робототехники. Дивноморск (2009)
- 4. MCO 2009: Всероссийская конференция Методы и средства обработки информации. Москва, МГУ (2009)
- 5. **ММРО-14:** 14-ая Всероссийская конференция Математические Методы Распознавания Образов. Суздаль (2009)
- 6. 52-я научная конференция МФТИ. Долгопрудный, МФТИ (2009)
- 7. Х
   Всероссийская
   научно-техническая
   конференция

   Нейроинформатика-2010.
   Москва, МИФИ (2010)
   Солона
   Солона</
- 8. **IJCNN 2010:** The International Joint Conference on Neural Networks. Spain, Barcelona (2010)

- ICANN 2010: The 20 International Conference on Artificial Neural Networks, Greece, Thessaloniki (2010)
- 10. **ICANN 2011:** The 21st International Conference on Artificial Neural Networks. Finland, Espoo (2011).
- 11. **ICCGI 2012:** The Seventh International Multi-Conference on Computing in the Global Information Technology. Italy, Venice (2012)
- 12. **ICANN 2012:** The 22 International Conference on Artificial Neural Networks. Switzerland, Lausanne (2012)

#### 1.8 Структура диссертации

Работа состоит из 5 глав, Заключения и Списка литературы.

Первая глава представляет собой введение, в котором дается описание проблемы, её состояние на данный момент и приводится обзор существующих методов ее решения.

В главе 2 показано, что при случайном поиске вероятность отыскания какого-либо минимума экспоненциально возрастает с ростом глубины этого минимума. В частности, при случайной инициализации нейросети она с наибольшей вероятностью конвергирует в состояние, соответствующее С несколько глобальному МИНИМУМУ. меньшей вероятностью она конвергирует в один из локальных минимумов, сравнимых по глубине с глобальным. А вероятность конвергенции в относительно неглубокие локальные минимумы экспоненциально мала. Получены ограничения на глубину минимумов, а также формулы для формы минимума и зависимости вероятности отыскания минимума от его глубины.

В главе 3 рассмотрен простейший вид трансформации, который производится путем возведения матрицы T в степень k (k = 2, 3, ...). На этой основе построен алгоритм минимизации DDK (Double Descent algorithm with parameter k). Алгоритм DDK был протестирован на задаче разбиения графа на две равные части (graph bipartitioning).

В главе 4 предложено трансформировать поверхность энергетического функционала смешивая исходную матрицу T с матрицей  $T^2$ . На этой основе построен алгоритм минимизации MM (Mix-Matrix algorithm). Алгоритм MM был протестирован на задаче о максимальном разрезе в графе (max-cut).

В **главе 5** методами статистической физики исследован вопрос об оценке нижнего предела энергии минимумов, и потому ее следует рассматривать как дополнение к главе 2.

В Заключении подводятся итоги о проделанных исследованиях.

Список цитированной литературы содержит 85 работ.

#### 2. ОБЛАСТЬ ПРИТЯЖЕНИЯ МИНИМУМА

В данной главе мы покажем, что с ростом глубины минимума растет радиус бассейна притяжения и, соответственно, возрастает вероятность его отыскания при случайном поиске. Для простоты изложения мы введем понятие глубины минимума, которую мы обозначим через  $\gamma \equiv |E|\sqrt{N}$ .

#### 2.1 Область притяжения минимума

Пусть  $S_0 = (s_1^{(0)}, s_2^{(0)}, ..., s_N^{(0)})$  - конфигурация, соответствующая некоторому минимуму  $E_0 = E(S_0)$ . Совокупность точек, находящихся на расстоянии *n* (по Хеммингу) от  $S_0$ , назовём *n*-окрестностью точки  $S_0$ . Тогда, объём области притяжения минимума (число всех точек пространства, стартуя с которых спускаемся в  $S_0$ ) задается выражением:

$$V = 2\sum_{n=1}^{N/2} \binom{N}{n} P_n \tag{2.1}$$

где  $P_n$  - вероятность попадания из *n*-окрестности в минимум  $S_0$ , которую мы оценим ниже.

Рассмотрим произвольную точку  $S_n$  в *n*-окрестности минимума  $S_0$ , полагая что точка  $S_n$  не является локальным минимумом. В конфигурации  $S_n$  имеется хотя бы один спин (*i*-ый спин), направление которого противоположно направлению действующего на него локального поля  $h_i(S_n)$ . Состояние такого спина неустойчиво и он должен развернуться вдоль поля  $h_i(S_n)$ . Переворот этого спина переведет систему в (n-1)-окрестность, если направление поля  $h_i(S_n)$  совпадает с направлением эталона  $s_i^{(0)}$ . В противном случае система перейдет в (n+1)-окрестность, т.е. отдалится от минимума. Вычислим вероятность перехода из *n*-окрестности в (n-1)-окрестность, которая совпадает с вероятностью события  $h_i(S_n)s_i^{(0)} > 0$ . Для этого представим *T* в виде:

$$T = T_0 + T_1, (2.2)$$

где

$$T_0 = r_0 \sigma_T S_0^+ S_0 \quad , \quad T_1 = T - T_0 \tag{2.3}$$

Вес  $r_0$  может быть найден из условия отсутствия корреляции между элементами матриц  $T_0$  и  $T_1$ . Рассчитав ковариацию матричных элементов и положив ее равной нулю, для величины  $r_0$  получим:

$$r_{0} = -\frac{E_{0} + \overline{T}\left(\overline{s_{i}^{(0)}}\right)^{2}}{1 - \left(\overline{s_{i}^{(0)}}\right)^{4}},$$
(2.4)

где  $\overline{s_i^{(0)}}$  - среднее значение величин спинов в точке  $S_0$ ,  $\overline{T}$  - среднее значение матричных элементов. Полагая  $\overline{s_i}$  и  $\overline{T}$  равными нулю, получаем:

$$r_0 = -E_0 \tag{2.5}$$

Это выражение устанавливает связь между  $r_0$  и глубиной минимума  $E_0$ . Дисперсии элементов матриц  $T_0$  и  $T_1$  имеют вид  $\sigma_0^2 = r_0^2 \sigma_T^2$  и  $\sigma_1^2 = \sigma_T^2 - \sigma_0^2$ , подтверждающий, что случайную матрицу T удалось представить как сумму двух независимых случайных матриц  $T_0$  и  $T_1$ . Кроме того, из (2.2) и (2.5) вытекает выражение  $S_0T_1S_0^+ = 0$ , показывающее, что вклад в энергию  $E_0$  от  $T_1$  строго равен нулю, т.е. минимум в  $S_0$  обусловлен исключительно вкладом от  $T_0$ .

Пользуясь соотношениями (2.2)-(2.5), выпишем выражение для локального поля, действующего на i-ый спин в конфигурации  $S_n$  из n-окрестности  $S_0$ :

$$h_i(S_n) = (N - 2n) \left| E_0 \right| s_i^{(0)} + \sum_{j \neq i} T_{1ij} s_j$$
(2.6)

В силу случайности матрицы  $T_1$ , сумма в (2.6) может рассматриваться как нормально распределённая величина с нулевым средним и дисперсией равной  $N\sigma_T^2$ . Т.е. произведение  $h_i(S_n)s_i^{(0)}$  представляет собой нормально распределенную величину со средним  $(N-2n)|E_0|$  и дисперсией  $N\sigma_T^2$ . Соответственно, для искомой вероятности перехода из *n*-окрестности в (n-1)-окрестность получим  $\Pr\{h_i(S_n)s_i^{(0)} > 0\} = \Phi(\gamma_n)$ , где:

$$\Phi(\gamma_n) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\gamma_n} e^{-\frac{1}{2}z^2} dz, \qquad \gamma_n = \gamma (1 - 2n / N)$$
(2.7)

 $\gamma = |E_0|\sqrt{N}$  - глубина минимума  $S_0$ . Тогда выражение для вероятности конвергенции из *n*-окрестности в минимум  $S_0$  примет вид:

$$P_n = \Phi(\gamma_n) \cdot \Phi(\gamma_{n-1}) \cdot \dots \cdot \Phi(\gamma_1).$$
(2.8)

Подставляя это выражение в (2.1), используя соотношение Стирлинга и переходя от суммирования к интегрированию по переменной x = n / N для объема области притяжения получим:

$$V = \sqrt{2N} \int_{0}^{0.5} \frac{e^{NF(x)}}{\sqrt{\pi x(1-x)}} dx,$$
 (2.9)

где

$$F(x) = -x \ln x - (1-x) \ln(1-x) + \int_{0}^{x} \ln \left[ \Phi(\gamma_{z}) \right] dz \quad , \qquad \gamma_{z} = \gamma (1-2z)$$

# (2.10)

Интеграл (2.9) оценим методом перевала. Приравняв производную *F*(*x*) к нулю, получим уравнение для седловой точки *x*<sub>0</sub>:

$$\frac{x_0}{1-x_0} = \Phi[\gamma(1-2x_0)]$$
(2.11)

и выражение (2.9) преобразуется к виду:

$$V = 2e^{NF(x_0)}$$
(2.12)

Для простоты в предэкспоненте выражения (2.12) опущен несущественный множитель, слабо отличающийся от единицы на всем интервале изменений величины  $\gamma$ .

фактически представляет собой радиус Величина области  $x_0$ притяжения минимума, поскольку объем области с точностью ДО несущественного коэффициента совпадает с величиной  $x_0^N$ . Анализ выражений (2.11)-(2.12) показывает, что с ростом глубины минимума у радиус области притяжения возрастает. Одновременно экспоненциально возрастает и объем области притяжения. Чтобы убедиться в этом, получим из (2.11)-(2.12) более простые оценочные выражения. Эксперимент показывает, что глубина среднестатистического минимума  $\overline{\gamma}$  отличается от глубины глобального минимума  $\gamma_0$  всего лишь на 10-15%. Кроме того, величины  $\overline{\gamma}$  и  $\gamma_0$  слабо меняются в зависимости от типа матрицы (см. Таблицу 2.1). Это позволяет для относительно глубоких минимумов  $(\overline{\gamma} < \gamma \leq \gamma_0)$  представить функцию  $x_0 = x_0(\gamma)$  в виде разложения В окрестности значения  $\overline{\gamma}$ , которое наиболее легко определяется В эксперименте. Ограничиваясь линейным членом разложения, получим оценочное выражение:

$$x_0 \approx \overline{x} + k \,\overline{x} \,(\gamma - \overline{\gamma}) \tag{2.13}$$

где  $\overline{x}$  - радиус притяжения для минимума среднестатистической глубины  $\overline{\gamma}$ , а k - коэффициент, слабо зависящий от типа матрицы (см. Таблицу 2.1). Аналогично, проводя в (2.12) разложение в окрестности величины  $\overline{\gamma}$ , для вероятности отыскания минимума  $P(\gamma) = V / 2^N$  получим:

$$P(\gamma) = \overline{P} e^{kN(\gamma - \overline{\gamma})}$$
(2.14)

где  $\overline{P} = P(\overline{\gamma})$  есть вероятность отыскания минимума среднестатистической глубины  $\overline{\gamma}$ .

В Таблице 2.1 приведены экспериментально определенные значения характерных параметров, позволяющие провести численную оценку для

полученных теоретических выражений. Взяв для примера некоторый среднестатистический минимум с  $\gamma = 1.36$ , характерный для SK-матриц, получаем  $\bar{x} \approx 0.384$  и  $k \approx 0.076$ . Это означает, что при случайном поиске большая часть точек, из которых система конвергирует в данный минимум, расположена в его *n*-окрестности, где  $n \approx 0.384N$ . Соответственно, вероятность отыскания минимума глубины  $\gamma$  задается выражением

$$P \approx \overline{P} e^{0.076N(\gamma - \overline{\gamma})}$$

Как видим, чем глубже минимум, тем больше вероятность его отыскания, причём зависимость экспоненциальная. Данная зависимость выполняется не строго, а лишь в статистическом смысле. При небольших значениях N её можно подтвердить экспериментально (см. рис. 2.1). При больших размерностях ( $N \ge 1000$ ) нет возможности проверить данную экспоненциальную зависимость, поскольку случайным поиском не удаётся попасть в один и тот же минимум более одного раза даже при достаточно большом ( $10^6 - 10^8$ ) числе стартов.



**Рис. 2.1.** Распределение числа попаданий в различные минимумы для одной случайной матрицы N = 100, число стартов  $10^6$ . По оси абсцисс – значение энергии минимумов,  $E_0$  соответствует энергии глобального минимума. По оси ординат – число попаданий. Число попаданий в глобальный минимум составило 11482. Полоса внизу отображает плотность найденных минимумов

	$\gamma_0$	$\overline{\gamma}$	<i>x</i> <sub>0</sub>	$\overline{x}$	$\ln(P_0)/N$	$\ln(\overline{P})/N$	k
2D EA	1.31	1.09	0.383	0.377	-0.125	-0.142	0.091
3D EA	1.38	1.15	0.385	0.379	-0.118	-0.138	0.088
SK	1.51	1.36	0.387	0.384	-0.111	-0.121	0.076

**Таблица 2.1**. Экспериментальные значения величин, характеризующих область притяжения, для матриц трёх типов моделей 2D EA, 3D EA и SK.

## 2.2 Форма минимума

Мы показали, что с ростом глубины минимума  $\gamma_0$  его ширина возрастает. Соответственно, вероятность отыскания этого минимума экспоненциально возрастает как  $P(\gamma_0) \sim \exp(kN\gamma_0)$ . Такую жесткую связь между глубиной минимума и вероятностью его отыскания несложно понять из следующих соображений. Вероятность отыскания глобального минимума в процессе случайного поиска есть не что иное, как отношение числа точек, находящихся в пределах его области притяжения, к общему числу точек  $(2^N)$  в *N*-мерном пространстве. В **Приложении А** показано, что форма энергетической поверхности вокруг минимума описывается выражениями:

$$\overline{E}_0(n) \approx E_0 \left(1 - \frac{2n}{N}\right)^2, \quad \sigma_0(n) \approx \frac{2}{N} \sqrt{\frac{2n}{N} \left(1 - \frac{n}{N}\right)}$$
(2.15)

где  $\overline{E}_0(n)$  и  $\sigma_0(n)$  - соответственно среднее значение и стандартное отклонение энергии в *n*-окрестности минимума (в точках  $S_0^{(n)}$ , отстоящих от  $S_0$  на расстоянии *n* по Хэммингу). Выражения (2.15) получены в асимптотическом пределе N>>1 из точных выражений, приведенных в Приложении А. Энергетическая поверхность (2.15) очень гладкая, поскольку дисперсия энергии в *n*-окрестности с ростом размерности стремится к нулю как  $\sigma \sim 1/N$ . Более того, все минимумы имеют одинаковую форму, задаваемую выражением (2.15). Как видно из (2.15), ширина минимума (на полувысоте) прямо пропорциональна его глубине  $E_0$ . Это внутри области И означает, ЧТО число точек притяжения экспоненциально возрастает с ростом глубины минимума.

#### 2.3 Ограничение на глубину минимума

Установим основные соотношения, связанные с ограничениями на глубину глобального (локального) минимума.

Выше (см. формулу (2.2)) нам удалось представить матрицу *T* как сумму двух независимых случайных матриц  $T_0$  и  $T_1$ . Следуя [72] (Kryzhanovsky (2008)) продолжим разложение (2.2). Пусть конфигурация  $S_1$ является экстремумом квадратичного функционала на основе матрицы  $T_1$ . Аналогично (2.2) представим  $T_1$  в виде  $T_1 = r_1 \sigma_T S_1^+ S_1 + T_2$ , определяя статвес  $r_1$ из условия нулевой ковариации элементов матриц  $S_1^+ S_1$  и  $T_2$ . Повторяя эту процедуру мы получаем представление матрицы в виде взвешенного разложения по внешним произведениям случайных конфигурационных векторов:

$$T = \sigma_T \sum_{m=0}^{\infty} r_m S_m^+ S_m , \quad \sum r_m^2 = 1$$
 (2.16)

Для такого типа матриц справедливы все выводы, полученные в [73] (Amit et al (1987)), а также в главе 5, на основании статистической физики. В частности, можно утверждать, что любой из векторов  $S_m$ , присутствующий в разложении матрицы T, будет минимумом функционала только в том случае, если его статвес  $r_m$  больше критического значения:

$$r_c = \frac{1}{2\sqrt{\alpha_c N}} \tag{2.17}$$

где  $\alpha_c \approx 0.138$  - критическое значение параметра загрузки [73] (Amit et al (1987)) (подробнее об ограничении на  $\alpha_c$  в случае квазихеббовской матрицы вида (2.16) см. главу 5). Это утверждение прежде всего относится к точке  $S_0$ , которая по определению является минимумом функционала (1.1), и для нее справедливы соотношения:

$$1 \ge r_0 \ge r_c$$
,  $-1 \le E_0 \le E_c$ ,  $E_c = -r_c$  (2.18)

Ограничение на  $r_0$  сверху очевидно: при  $r_0 \rightarrow 1$  имеем  $T = T_0$ , т.е. матрица T образована как внешнее произведение вектора  $S_0$ , и функционал E(S) имеет единственный минимум в точке  $S_0$ . Этот предел не представляет интереса, поскольку отыскание глобального минимума не представляет труда. В подавляющем большинстве минимизационных приложений реализуется ситуация  $r_0 \sim r_c$  и  $E_0 \sim E_c$ , когда вероятность отыскания глобального минимума чрезвычайно мала. Именно эту ситуацию мы и будем анализировать.

Все полученные выше выводы получены на примере конфигурации  $S_0$ , соответствующей глобальному минимуму. Однако, они справедливы для любого экстремума  $S_m$  функционала E(S): для минимумов справедливы ограничения  $1 \ge r_m \ge r_c$  и  $E_c \ge E_m \ge -1$ , ограничения для максимумов имеют вид  $r_c \le -r_m \le 1$  и  $|E_c| \le E_m \le 1$ .

#### 2.4 Выводы по главе 2

В данной главе мы получили несколько результатов, которые будут использоваться для построения алгоритмов случайного поиска, описанных ниже в диссертации.

Во-первых, радиус области притяжения минимума напрямую зависит от его глубины: чем глубже минимум, тем больше его радиус области притяжения. На практике для глубоких минимумов эту зависимость можно приближённо считать линейной.

Во-вторых, показано, что вероятность отыскания минимума растёт экспоненциально с ростом его глубины. Поэтому глубокие минимумы находятся чаще, а мелкие реже.

В-третьих, мы установили, что глубина минимума  $|E_0|$  всегда больше критического значения  $|E_c|$ . Это также означает, что точка *S* скорее всего не будет минимумом, если её глубина меньше некоторого критического значения  $|E_c|$ .

Отсюда вытекает направленность наших усилий по повышению эффективности алгоритма случайного поиска: трансформацию энергетической поверхности необходимо произвести таким образом, чтобы увеличить глубину глобального минимума и, тем самым, увеличить вероятность его отыскания, и наоборот, уменьшить глубину мелких минимумов и таким образом полностью их вытеснить.

# 2.5 Приложение А. Форма энергетической поверхности в покрестности

Рассмотрим произвольную точку  $S_0 = (s_1^{(0)}, s_2^{(0)}, ..., s_N^{(0)})$  в *N*-мерном конфигурационном пространстве. Набор точек  $\Omega = \{S_0^{(n)}\}$  таких что  $S_0^{(n)}$  расположено на расстоянии *n* бит от  $S_0$  называется *n*-окрестностью точки  $S_0$ . Покажем что при  $N \rightarrow \infty$  дсперсия энергии в *n*-окрестности пренебрежимо мала и энергетическая поверхность вокрут точки  $S_0$  определяется средней энергией в *n*-окрестности.

Компоненты вектора  $S_0^{(n)}$ , принадлежащие *n*-окрестности, и компоненты  $S_0$  связаны соотношением:

 $s_i^{(n)} = a_i^{(n)} s_i^{(0)} \forall i = \overline{1, N},$ 

где  $a_n$  - вектор, такой что его N - n компонент равны +1, а другие n компонент -1. Используя эти обозначения, энергия в точке  $S_0^{(n)}$  равна:

$$E_0(n) = -c \sum_i \sum_j T_{ij} s_i^{(0)} s_j^{(0)} a_i^{(n)} a_j^{(n)} \chi_{ij},$$
(A1)

где  $\chi_{ij}$  - единичный антисимметричный тензор.

Далее, сумма энергий и сумма квадратов энергий по множеству всех точек *n*-окрестности описываются выражениями:

$$\sum_{\Omega} E_0(n) = -c \sum_{i=1}^N \sum_{j=1}^N K_{ij} T_{ij} s_i^{(0)} s_j^{(0)}$$
(A2)

$$\sum_{\Omega} E_0^2(n) = -c \sum_{i,j,k,l}^N K_{ijkl} T_{ik} T_{jl} s_i^{(0)} s_k^{(0)} s_j^{(0)} s_l^{(0)},$$
(A3)

где

$$K_{ij} = \sum_{\Omega} a_i a_j \chi_{ij}, K_{ijkl} = \sum_{\Omega} a_i a_j a_k a_l \chi_{ik} \chi_{jl}$$
(A4)

Суммируя (A4) по сфере  $\Omega = \{S_n\}$  и подставляя результат в (A2) и (A3), мы получаем среднее значение и дисперсию энергии в *n*-окрестности:

$$\overline{E}_0(n) = E_0 \frac{(N-2n)^2 - N}{N(N-1)},$$
(A5)

$$\sigma_0^2(n) = \frac{8n(N-n)}{N(N-1)(N-2)(N-3)} \Big[ A \Big( 1 - E_0^2 \Big) + B \sigma_{hs}^2 \Big],$$
(A6)

где  $E_0 = -\sum h_i s_i^{(0)}$  - энергия системы в точке  $S_0$ ,  $h_i$  - локальное поле, действующее на *i* -й спин конфигурации  $S_0$ ,  $\sigma_{hs}$  - стандартное отклонение величины  $h_i s_i^{(0)}$ , и мы использовали следующие обозначения:

$$A = \frac{4(n-1)(N-n-1)}{N(N-1)},$$
(A7)

$$B = 2N \Big[ (N - 2n)^2 - (N - 2) \Big].$$
 (A8)

Согласно (А5) среднее значение энергии в *n*-окрестности ведёт себя параболически. При n = 0 среднее значение равно  $E_0$ ; с увеличением *n* средняя энергия  $\overline{E}_0(n)$  растёт как если бы  $S_0$  была окружена воронкой с параболическими стенками. При  $n = \frac{1}{2}(N - \sqrt{N})$  величина  $\overline{E}_0(n)$  обнуляется и при n = N/2 парабола достигается своего пика. Дальнейшее поведение кривой полностью симметрично: с ростом *n* средняя энергия падает параболически вплоть до энергии  $E_0$  при n = N когда конфигурация переходит в своё зеркальное отражение  $S_0^* = -S_0$ .

Выражения (A5)-(A6) справедливы для любой точки конфигурационного пространства. Однако  $\overline{E}_0(n)$  описывает более точно форму энергетической поверхности вокруг минимума: как следует из (A6),

дисперсия энергии минимальна когда  $S_0$  является точкой глобального минимума, поскольку  $\sigma_{hs}$  минимальна и  $E_0^2$  максимальна. Энергетическая поверхность описываемая средним  $\overline{E}_0(n)$  очень гладкая, поскольку неровность исчезает с ростом N как  $\sigma_0(n) \square N^{-1}$ .

Нетрудно посчитать  $\sigma_{hs}$  для любой точки пространства. Однако когда мы считаем  $\sigma_{hs}$  для минимума  $S_0$ , мы должны принять во внимание, что  $h_i s_i^{(0)} > 0$ ,  $\forall i = \overline{1, N}$ . С учётом этого, мы имеем  $2N^3 \sigma_{hs}^2 \Box 1$  (дисперсия  $h_i s_i^{(0)}$  в минимуме в два раза меньше чем в произвольной точке). Используя это соотношение и оставляя только первые члены в (A5)-(A6), которые не убывают как  $\Box O(N^{-1})$ , мы получаем приближённое выражение (2.15).

Суммируя (A2)-(A3) по всем *n*-окрестностям ( $n = \overline{0,N}$ ), т.е. по всему пространству, мы получаем тривиальный результат: среднее значение энергии равно нулю ( $\overline{E_0(n)} = 0$ ), а средний квадрат энергии определяется только средним квадратом матричных элементов ( $\overline{E_0^2(n)} = 2c^2N(N-1)\overline{T_{ij}^2}$ ), который совпадает с дисперсией матричных элементов при условии нулевого среднего  $\overline{T_{ij}}$ . Это объясняет значение нормализовочной константы в функционале энергии: благодаря этой нормализации величина энергии не зависит от сорта матрицы, поэтому мы можем сравнивать различные матрицы одного и того же типа, а также различные типы матриц.

#### **3.** АЛГОРИТМ С ВОЗВЕДЕНИЕМ МАТРИЦЫ В СТЕПЕНЬ (DDK)

эффективность процедуры случайного поиска пытаются Обычно повысить видоизменяя динамику спуска по поверхности, описываемой функционалом. В отличие от такого подхода, в настоящей работе предлагается изменять не динамику спуска ПО поверхности, a трансформировать саму поверхность так, чтобы увеличился радиус области притяжения глобального минимума (равно как и других минимумов, сравнимых с глобальным по глубине).

В данной главе мы рассмотрим простейший вид трансформации, возведения Т который производится путем матрицы В степень k (k = 2, 3, ...). Предлагаемый подход оказался весьма продуктивным: трансформация поверхности приводит к экспоненциальному ПО Nувеличению вероятности отыскания глобального минимума и сильному смещению спектра находимых минимумов в глубокую сторону.

Строгое обоснование эффективности предлагаемого алгоритма (DDKалгоритма) мы приведем только для «случайных» матриц, элементы которых получены как случайные независимые переменные (матрицы модели Эдвардса-Андерсона, матрицы с равномерным или нормальным распределением матричных элементов (модель Шеррингтона-Киркпатрика) и т.д.). Применение алгоритма для другого типа матриц будет эвристическим.

#### 3.1 Стандартный алгоритм случайного поиска (SRS)

Ещё раз повторим постановку задачи и стандартный алгоритм её решения.

Постановка задачи бинарной минимизации такова: задана  $N \times N$ матрица  $T_{ij}$ , следует найти N-мерный конфигурационный вектор  $S_m = (s_1^{(m)}, s_2^{(m)}, ..., s_N^{(m)}), \quad s_i^{(m)} = \pm 1, \quad i = \overline{1, N},$  доставляющий минимум функционалу энергии  $E_1(S)$ :

$$E_1(S) = -c_1 \sum_{i=1}^{N} \sum_{j=1}^{N} T_{ij} s_i s_j \quad , \qquad c_1 = \frac{1}{\sigma_T N(N-1)}$$
(3.1)

где  $c_1$  - нормировочная константа;  $\sigma_T$  - стандартное отклонение матричных элементов  $T_{ij}$ . Без ограничения общности будем полагать, что  $T_{ij}$  симметричная матрица с нулевой диагональю ( $T_{ij} = T_{ji}$ ,  $T_{ii} = 0$ ,  $\forall i, j = \overline{1, N}$ ) и со средним нулевым значением элементов ( $\overline{T_{ij}} = 0$ ).

За основу процедуры минимизации примем асинхронную динамику модели Хопфилда: вычисляется величина локального поля  $h_i = -\partial E_1(S) / \partial s_i$ , действующего на произвольно выбранный *i*-й спин:

$$h_i = \sum_{j \neq i}^N T_{ij} s_j \tag{3.2}$$

Если спин находится в неустойчивом состоянии ( $h_i s_i < 0$ ), то состояние спина обновляется в соответствии с решающим правилом  $s_i = \operatorname{sgn} h_i$ . Эта процедура последовательно применяется ко всем нейронам до тех пор, пока сеть не конвергирует в устойчивое состояние  $S_m$ . В каждый момент времени обновляется состояние только одного спина. Уменьшение энергии при каждом перевороте неустойчивого спина гарантирует, что процесс, описываемый асинхронной динамикой, за конечное число шагов приводит систему в устойчивое положение с минимумом энергии.

```
Algorithm SRS (Standard Random Search)
Begin
     Step 1. Random Initialization
     Initialize configuration S = (s_1, s_2, ..., s_N) at random
     Step 2. Descent over landscape E_{\!\!1} from S to minimum S_{\!\!m}
     Calculate h_i = \sum T_{ii} s_i for all i = \overline{1, N}
     while there are unstable spins ( h_i s_i < 0 )
           for each spin in {\it S}
                if h_i s_i < 0 then
                      S_i = -S_i
                      Refresh h_i = h_i + 2T_{ii}s_i for all j \neq i
                 end if
           end for
     end while
     Repeat from Step 1 until the minimum of a required depth
     is reached or until the uptime ends.
End Algorithm
```

**Рис. 3.1**. Стандартный алгоритм случайного поиска SRS, основанный на асинхронной нейросетевой динамике.

Алгоритм случайного поиска состоит в многократном повторении спусков из различных случайных начальных конфигураций, осуществляемых до тех пор, пока не будет найден минимум приемлемой глубины (возможно и глобальный минимум). Формальный вид стандартного алгоритма случайного поиска представлен на рис. 3.1. В дальнейшем мы будем обозначать его аббревиатурой SRS (Standard Random Search) и сравнивать с ним эффективность предлагаемого нами алгоритма.

Несмотря на простоту алгоритма SRS, спуск с одной случайной конфигурации требует приблизительно  $O_{SRS} \approx 2N^2$  операций для матриц модели Шеррингтона-Киркпатрика и  $O_{SRS} \approx 8N$  операций для матриц двумерной модели Эдвардса-Андерсона, поскольку в этой модели лишь 4N ненулевых матричных элементов. Более того, количество локальных N экспоненциально увеличивается, МИНИМУМОВ с ростом а значит, обнаружения экспоненциально уменьшается вероятность глобального Эффективность случайного минимума. процедуры обычно поиска трех параметров: характеризуют ОДНИМ ИЗ вероятность отыскания глобального минимума, скорость отыскания минимума в заданном интервале энергий, среднее значение глубины находимых минимумов.

#### 3.2 Базовые соотношения

Прежде чем перейти к вопросу о трансформации поверхности, напомним основные соотношения, связанные с глубиной глобального (локального) минимума, на которых будут базироваться все последующие выводы.

Пусть  $S_0 = (s_1^{(0)}, s_2^{(0)}, ..., s_N^{(0)})$  - конфигурация, соответствующая глобальному минимуму  $E_0 = E_1(S_0)$ . Как было показано в главе 2, случайную матрицу T можно представить как сумму двух независимых случайных матриц  $T_0$  и  $T_1$ :

$$T = T_0 + T_1, (3.3)$$

где

$$T_0 = r_0 \sigma_T S_0^+ S_0 \quad , \quad T_1 = T - T_0, \tag{3.4}$$

причём вес  $r_0$  напрямую связан с энергией в точке  $S_0$ :

$$r_0 = -E_0 \tag{3.5}$$

Дисперсии элементов матриц  $T_0$  и  $T_1$  имеют вид  $\sigma_0^2 = r_0^2 \sigma_T^2$  и  $\sigma_1^2 = \sigma_T^2 - \sigma_0^2$ . Кроме того, из (3.4)-(3.5) вытекает выражение  $S_0 T_1 S_0^+ = 0$ , показывающее, что вклад в энергию  $E_0$  от  $T_1$  строго равен нулю, т.е. минимум в  $S_0$  обусловлен исключительно вкладом от  $T_0$ .

Первое соотношение, которое было получено в главе 2, состоит в том, что чем глубже минимум  $E_0$ , тем больше статвес  $r_0$  примеси конфигурации  $S_0$  в исходной матрице T и тем больше вероятность его отыскания. Вероятность отыскания этого минимума экспоненциально возрастает с глубиной минимума как  $P(E_0) \sim \exp(kN\sqrt{N}|E_0|)$ .

Второе соотношение состоит в том, что точка  $S_0$  может быть минимумом только в случае  $r_0 \ge r_c$ , т.е. глубина минимума  $|E_0|$  всегда больше критического значения  $|E_c|$ . Отсюда вытекает направленность наших усилий по повышению эффективности алгоритма случайного поиска: трансформацию энергетической поверхности (3.1) необходимо произвести таким образом, чтобы увеличить глубину глобального минимума и, тем самым, увеличить вероятность его отыскания.

### 3.3 Трансформация поверхности

Трансформировать описываемую квадратичной формой  $E_1(S)$  поверхность можно только трансформируя матрицу, на которой она построена. Подставим в выражение (3.1) матрицу  $M = (1 - z)T + zT^k$ , где  $T^k$  - матрица, получаемая возведением матрицы T в степень k с последующим обнулением диагональных элементов. Изменяя параметр z от 0 до 1 перейдем от матрицы T к матрице  $M = T^k$ . Соответственно, поверхность, описываемая функционалом  $E_1(S)$ , трансформируется в поверхность, описываемую функционалом  $E_k(S)$ :

$$E_k(S) = -c_k \sum_{i=1}^{N} \sum_{j=1}^{N} M_{ij} s_i s_j \quad , \quad c_k = \frac{1}{\sigma_M N(N-1)}$$
(3.5)

где  $\sigma_M$  - стандартное отклонение элементов матрицы M. Все рассуждения проведем на примере трансформации глобального минимума. Очевидно, что при трансформации поверхности глобальный минимум сдвигается в пространстве, одновременно изменяются его глубина и ширина области притяжения. Ниже мы покажем, что при относительно небольших значениях параметра k ( $2 \le k \le 5$ ) указанная трансформация приводит к значительному увеличению глубины минимума при относительно небольшом сдвиге.

Соответственно, предлагается следующий двухэтапный алгоритм минимизации. На первом этапе осуществляется спуск по поверхности  $E_k(S)$  и находится конфигурация  $S_m^{(k)}$ , доставляющая минимум функционалу  $E_k(S)$ . На втором этапе проводится коррекция: из точки  $S_m^{(k)}$  по поверхности  $E_1(S)$  спускаемся в ближайший минимум  $S_m$  функционала  $E_1(S)$ . Алгоритм
спуска по поверхности  $E_k(S)$  в точности повторяет описанный выше: рассчитывается локальное поле  $h_i^{(k)} = -\partial E_k(S) / \partial s_i$  на *i*-м спине:

$$h_i^{(k)} = \sum_{j \neq i}^N M_{ij} s_j$$
(3.6)

и при  $h_i^{(k)} s_i < 0$  состояние спина обновляется по решающему правилу  $s_i = \operatorname{sgn} h_i^{(k)}$ .

```
Algorithm DDK (Double Descent with parameter k=2,3,...)
begin
      Step 1. Random Initialization
      Initialize configuration S = (s_1, s_2, ..., s_N) at random
      Step 2. Descent over landscape E_{\!\!k} from S to minimum S_{\!\!m}^{(k)}
      Calculate h_i^{(k)} = \sum M_{ij} s_i for all i = \overline{1, N}
      while there are unstable spins ( h_i^{(k)}s_i < 0 )
            for each spin in {\it S}
                  if h_i^{(k)}s_i < 0 then
                         S_i = -S_i
                        Refresh h_i^{(k)} = h_i^{(k)} + 2M_{ii}s_i for all j \neq i
                  end if
            end for
      end while
      Step 3. Correction: descent over landscape E_1 from S_{\rm m}^{(k)} to minimum S_{\rm m}
      Calculate h_i = \sum T_{ij} s_j for all i = \overline{1,N} at configuration S = S_m^{(k)}
      while there are unstable spins ( h_i s_i < 0 )
            for each spin in \,S\,
                  if h_i s_i < 0 then
                         S_i = -S_i
                        Refresh h_i = h_i + 2T_{ij}s_i for all j \neq i
                  end if
            end for
      end while
end Algorithm
```

Рис. 3.2. Алгоритм DDK случайного поиска, основанный на двухэтапном спуске.

На основе предлагаемого двухэтапного спуска мы реализовали алгоритм случайного поиска, формальный вид которого представлен на рис. 3.2. В дальнейшем мы будем обозначать его аббревиатурой DDK (Double Descent Algorithm с параметром K = 2,3,...) и сравнивать его эффективность со стандартным алгоритмом случайного поиска SRS. Во избежание недоразумений отметим, что при k = 1 трансформированный функционал  $E_k(S)$  совпадает с исходным  $E_1(S)$ , а предлагаемый алгоритм DDK в этом случае есть не что иное, как стандартный алгоритм случайного поиска SRS  $(DD1 \equiv SRS)$ .

Количество операций, выполняемых алгоритмом DDK примерно в два раза больше чем у алгоритма SRS, т.е.  $O_{DDK} \approx 2O_{SRS}$ , однако мы также должны помнить что некоторое время также приходится тратить на возведение матрицы в степень (порядка  $N^3$  операций для полносвязных матриц).

### 3.4 Обоснование алгоритма DDK

Обоснование предложенного алгоритма проведем на примере случая k = 2. Покажем, что в результате трансформации поверхности глобальный минимум становится заметно глубже, а его сдвиг в пространстве невелик.

### 3.4а) Углубление минимумов

Сначала убедимся, что трансформация поверхности приводит к углублению минимума. Рассмотрим энергию  $E_{20} = E_2(S_0)$  в точке  $S_0$ . Следуя (3.3) матрицу  $M = T^2$  представим в виде  $M = T_0^2 + T_1^2 + (T_0T_1 + T_1T_0)$ . Тогда с учетом соотношений  $S_0T_1S_0^+ = 0$  и  $\sigma_M = \sqrt{N}\sigma_T^2$  из (3.5) получим:

$$E_{20} = -\sqrt{N}E_0^2 + \frac{1}{N^{5/2}\sigma_T^2} \sum_{i=1}^N \sum_{j \neq i}^N (T_1^2)_{ij} s_i^{(0)} s_j^{(0)}$$
(3.7)

Из (3.7) следует, что в пределе N >> 1 энергию  $E_{20}$  можно рассматривать как нормально распределенную величину со средним  $\overline{E}_{20}$  и относительно малым стандартным отклонением  $\sigma_E$ :

$$\overline{E}_{20} = -\sqrt{N}E_0^2$$
,  $\sigma_E = (1 - r_0^2) / N$  (3.8)

Поскольку  $\overline{E}_{20} / E_0 = r_0 \sqrt{N} \ge r_c \sqrt{N} \approx 1.35$ , то следует ожидать, что в результате произведенной трансформации поверхности минимум станет глубже ( $E_{20} < E_0$ ). Вероятность такого события описывается выражением:

$$\Pr\{E_{20} < E_0\} = \frac{1}{2}(1 + erf \eta)$$
(3.9)

где

$$\eta = \frac{E_0 - \overline{E}_{20}}{\sqrt{2}\sigma_E} = \frac{r_0 N(r_0 \sqrt{N} - 1)}{\sqrt{2}(1 - r_0^2)}$$
(3.10)

Из соотношения  $r_0 \ge r_c$  следует  $\eta \ge 0.3\sqrt{N}$ . Следовательно, с ростом N вероятность углубления  $\Pr\{E_{20} < E_0\}$  асимптотически стремится к единице. Иными словами, трансформация поверхности с подавляющей вероятностью приводит к существенному увеличению глубины минимума (в среднем в 1.35 раз), что в соответствии с выше сказанным должно приводить к экспоненциальному по N возрастанию вероятности отыскания глобального минимума.

Отметим здесь, что  $\overline{E}_{20} = -\sqrt{N}E_0^2$ . Из этого выражения видно, что минимум функционала  $E_2(S)$  достигается как в конфигурации  $S_0$ , соответствующей минимуму ( $E_0 < 0$ ) функционала  $E_1(S)$ , так и в конфигурации  $S_0^*$ , соответствующей его максимуму ( $E_0 > 0$ ). Этот факт подтверждается в эксперименте.

### 3.4б) Сдвиг минимума

Теперь оценим насколько сдвинется минимум при трансформации поверхности. Среднюю величину сдвига можно представить в виде  $d_k = NP$ , где

$$P = \Pr\{s_i^{(0)} h_i^{(k)} < 0 \mid h_i s_i^{(0)} > 0\}$$
(3.11)

- вероятность того, что направления спина  $s_i^{(0)}$  и локального поля  $h_i^{(k)}$  в конфигурации минимума  $S_0$  не совпадают. Для рассматриваемого здесь случая k = 2 из (3.6) для величины  $s_i^{(0)}h_i^{(2)}$  получим:

$$s_i^{(0)}h_i = r_0\sigma_T N + \xi_i \tag{3.12}$$

$$s_i^{(0)} h_i^{(2)} = r_0^2 \sigma_T^2 N^2 + r_0 N \xi_i + \xi_{2i}$$
(3.13)

где

$$\xi_i = \sum_{j \neq i} (T_1)_{ij} \, s_i^{(0)} s_j^{(0)} \tag{3.14}$$

$$\xi_{2i} = \sum_{j \neq i} \left( T_1^2 \right)_{ij} s_i^{(0)} s_j^{(0)}$$
(3.15)

- нормально распределенные величины с нулевым средним и стандартным отклонением  $\sigma_{\xi_i} = \sigma_{T_1} \sqrt{N/2}$  и  $\sigma_{\xi_{2i}} = \sigma_{\xi_i}^2$ . С учетом этого выразим вероятность (3.11) через функцию ошибок

$$P = \frac{1}{P_0 \sqrt{\pi}} \int_0^{+\infty} e^{-(x-\gamma)^2} erfc(\sqrt{2\gamma}x) dx$$
(3.16)

где  $P_0 = 1 + erf \gamma$  и  $\gamma = \sqrt{N}r_0$ .

Как видим, сдвиг минимума  $d_2 = NP$  невелик: с учетом (3.16) мы получаем, что  $d_2 \le 0.026N$ . Можно показать, что для случая k = 3 сдвиг оказывается ещё меньше.

Аналогично можно показать, что дисперсия сдвига равна:

$$\sigma_{dk}^2 = NP(1-P) \tag{3.17}$$

Пренебрегая множителем (1 - P), получаем, что  $\sigma_{dk}^2 \approx d_k$ . То есть, в то время как среднее значение сдвига пропорционально размерности задачи N, стандартное отклонение пропорционально  $\sqrt{N}$ .

## 3.4в) Выводы и их проверка

Из выражений (3.8)-(3.17) можно сделать следующие выводы. Трансформация поверхности с подавляющей вероятностью приводит к углублению минимумов и, как следствие, к увеличению вероятности их отыскания. Причем, средняя величина углубления  $\overline{E}_{20} / E_0 = |E_0| \sqrt{N}$  тем больше, чем больше их изначальная глубина  $|E_0|$ . Иными словами, глубокие минимумы становятся еще более глубокими и вероятность их отыскания становится еще больше, а мелкие минимумы становятся еще мельче (или вовсе исчезают) и вероятность их отыскания становится еще меньше. Исходя из сказанного следует ожидать, что спектр минимумов, находимых с помощью предлагаемого алгоритма, существенно сдвинется в сторону глобального минимума, а вероятность отыскания последнего существенно возрастет. Рис. 3.3 показывает наблюдаемое углубление глобальных минимумов. Как видим, глобальные минимумы становятся глубже для k = 2,3,4 и даже для k = 5, однако с увеличивающейся дисперсией. Наибольшее среднее углубление наблюдается для k = 3.



**Рис. 3.3** Углубление минимума и стандартное отклонение энергии  $E_k(S_0)$  при трансформации поверхности при возведении матрицы в степень k = 2, 3, ..., 7. Матрицы двумерной модели Эдвардса-Андерсона размера N = 100, 196, 484.



**Рис. 3.4**. Слева: иллюстрация того, как трансформация энергетической поверхности функционала приводит к углублению минимума на величину  $\Delta E$ , а также к небольшом смещению в пространстве на величину *d*.Справа: спектральная плотность минимумов функционалов  $E_1(S)$  и  $E_2(S)$ . Квадратами отмечены энергии  $E_1(S_0)$  и  $E_2(S_0)$  в конфигурации  $S_0$ , соответствующей глобальному минимуму функционала  $E_1(S)$ .

Иллюстрация трансформации энергетической поверхности приведена на рис. 3.4 слева.

Поскольку углубление минимумов должно сопровождаться увеличением вероятности их отыскания, то спектр минимумов трансформированного функционала должен быть сдвинут в глубокую сторону. В подтверждение сказанному на рисунке 3.4 справа показаны спектральные плотности минимумов двух функционалов  $E_1(S)$  и  $E_2(S)$ , полученные при N = 100 для двумерной модели Эдвардса-Андерсона (2D EA) модели. Спектральная определена следующим образом: N(E)dE – число N(E)плотность локальных минимумов, найденных на интервале с энергией [E, E + dE], нормированное на общее число обнаруженных минимумов. Как видим, спектр функционала  $E_2(S)$  сдвинут влево относительно спектра исходного функционала  $E_1(S)$ , а минимум в точке  $S_0$  стал заметно глубже. Число обнаруженных минимумов функционала  $E_2(S)$  приблизительно в 2 раза меньше, чем у функционала  $E_1(S)$ . Это показывает, что мы добились желаемого: в результате трансформации поверхности глубокие минимумы стали глубже и детектируются с большей частотой, а мелкие детектируются с меньшей частотой либо вовсе исчезли.



**Рис. 3.5**. Средний сдвиг глобальных минимумов, вызванный трансформацией энергетической поверхности. Кривые получены экспериментально для 2D EA матриц размера N = 100,196,484. Для SK модели поведение сдвигов аналогичное.

Вызываемый трансформацией сдвиг минимума в пространстве относительно невелик: из (3.16) следует, что наименьшие сдвиги ожидаются для наиболее глубоких минимумов. Так, при N = 100, измеряемый в эксперименте сдвиг локальных минимумов на разных матрицах 2D EA модели варьировался от 0 до 6 бит, а его среднее значение составляло  $d \sim 3.5$  бит, что находится в хорошем согласии с (3.16). Рис. 3.5 показывает связь между сдвигом и степенью трансформации k (напомним что дисперсия сдвига совпадает со средним значением сдвига). Видно, что наименьший сдвиг наблюдается при k = 3.

Как видим, все предсказания теории находятся в согласии с экспериментом и являются хорошим обоснованием предлагаемого алгоритма DDK. В последующих разделах мы приведем данные по применению этого алгоритма для матриц разного типа.

### 3.5 Эффективность алгоритма DDK

Чтобы оценить эффективность предложенного алгоритма, мы провели несколько экспериментов. В качестве тестовых использовалось несколько типов матриц. Наши компьютерные симуляции включали в себя по 10<sup>6</sup> стартов для каждого эксперимента с динамикой Хопфилда и по 10<sup>4</sup> стартов для динамики Хоудайера-Мартина (см. дальше). Результаты были усреднены по ансамблю из 100 матриц каждого типа.

3.5а) Результаты для матриц двумерной модели Эдвардса-Андерсона (2D EA)

Алгоритм тестировался на матрицах двумерной модели Эдвардса-Андерсона размерностью N = 100. Данный тип матриц выбран по двум причинам. Во-первых, стандартная процедура случайного поиска для данного типа матриц малоэффективна – при N = 100 вероятность отыскания глобального минимума меньше  $2 * 10^{-6}$ . Во-вторых, для этих матриц методом ветвей и границ [33] (Liers et al (2004)) можно найти глобальный минимум  $S_0$  (мы использовали данные взятые с сайта http://www.informatik.uni-koeln.de/spinglass/).

Эффективность алгоритма DDK будем определять по трем параметрам: вероятность отыскания глобального минимума  $P_0$ , разница между средней энергией локальных минимумов  $\overline{E}_m$  и энергией глобального минимума  $E_0$ , выражаемая безразмерной величиной отклонения  $\delta E = (\overline{E}_m - E_0) / E_0$ , и вероятность попадания в интервал энергий  $[E_0, 0.99E_0]$ .

Проверка эффективности алгоритма DDK проводилась при k от 2 до 7. Прежде всего мы следили за тем, как с ростом параметра трансформации k изменяется спектр локальных минимумов функционала  $E_1(S)$ , находимых алгоритмом DDK. Обработка данных позволила получить плотность вероятности P(E), где P(E)dE - вероятность отыскания минимумов в интервале энергий [E, E + dE]. На рис. 3.6 показаны кривые плотности вероятности, полученные для 2D EA модели при N = 100, энергия глобального минимума принята за -1. Как видим, с ростом параметра трансформации k спектр находимых минимумов смещается в сторону глобального минимума. Более того, становится заметно отличной от нуля даже вероятность отыскания глобального минимума.

Помимо этого, исследовали как В зависимости ΜЫ OT типа используемого алгоритма поиска изменится распределение локальных минимумов в пространстве вокруг глобального минимума. Распределение задается достигаемой алгоритмом DDK вероятностью  $W_k = W_k(d)$ , с которой локальный минимум будет найден на расстоянии d (по Хеммингу) от глобального минимума. Абсолютные значения величины  $W_k = W_k(d)$  не информативны. Поэтому на рисунке 3.7 мы привели величину  $W_k$ , нормированную на вероятность  $W_{SRS} = W_{SRS}(d)$ , получаемую стандартным алгоритмом SRS. Как видим, использование алгоритма DDK значительно вероятность отыскания локального вблизи увеличивает минимума

глобального. В частности, с ростом k более чем на три порядка возрастает вероятность отыскания глобального минимума (d = 0).



**Рис. 3.6**. P(E) - плотность вероятности отыскания минимумов функционала  $E_1(S)$ , получаемая стандартным алгоритмом случайного поиска SRS и алгоритмом двойного спуска DDK при k = 2, 3, 4, 5.



**Рис. 3.7**.  $W_k = W_k(d)$  - вероятность отыскания минимумов на расстоянии d от глобального минимума, получаемая алгоритмом DDK при k = 2, 3, 4, 5. Вероятность  $W_k$ , нормирована на величину  $W_{SRS} = W_{SRS}(d)$ , получаемую стандартным алгоритмом случайного поиска SRS. 2D EA N = 100

Результаты численных экспериментов сведены в Таблице 3.1. В первой колонке приведены данные для стандартного алгоритма SRS - спуск по поверхности  $E_1(S)$ , в последующих колонках - данные для нового алгоритма k = 2 - 5). Выделены DDK (при три характеристики ДЛЯ оценки эффективности алгоритмов: в первой строке приведена величина отклонения  $\delta E = (\bar{E}_m - E_0) / E_0$ ; во второй строке - вероятность попадания в близкий к глобальному минимуму интервал энергий [E<sub>0</sub>, 0.99E<sub>0</sub>]; в третьей строке - вероятность попадания в глобальный минимум  $P_0$ .

**Таблица 3.1.** Сравнение эффективности алгоритма DDK с результатами стандартного алгоритма случайного поиска SRS, проведенное для 2D EA модели при размерности  $N = 10 \times 10$ .

	SRS	DD2	DD3	DD4	DD5
$\delta E = \left(\overline{E}_m - E_0\right) / E_0$	16%	11%	6%	10%	5%
$\Pr[E_0, 0.99E_0]$	$2.8*10^{-5}$	2.3*10 <sup>-3</sup>	$1.1*10^{-2}$	5.6*10 <sup>-3</sup>	$2.5*10^{-2}$
$P_0$	$2.4*10^{-6}$	3.8*10 <sup>-4</sup>	$2.1*10^{-3}$	$1.2*10^{-3}$	$4.1*10^{-3}$

Видно, что результаты для k = 5 мало отличаются от результатов для k = 3. При k > 5 результаты становятся хуже, поэтому дальнейший рост показателя матрицы k не целесообразен: эффективность алгоритма с ростом k уменьшается, а вычислительная сложность повышается.

#### 3.56) Результаты для матриц модели Шеррингтона-Киркпатрика (SK)

Матрицы данного типа полносвязные. В отличие от матриц модели Эдвардса-Андерсона, для них относительно высока вероятность (порядка 1% при *N* = 100) найти глобальный минимум с помощью обычного поиска. Однако было интересно проверить насколько эффективен предложенный метод при работе и с этими матрицами тоже.

Таблица 3.2 показывает результаты трансформации поверхности для матриц данного типа. Мы работали с размерностями N = 100 и N = 512 Видно, что трюк с возведением также увеличивает эффективность минимизации и для этого типа матриц. Действительно, если мы используем SRS, вероятность нахождения глобального минимума чуть меньше процента для N = 100, в то время как возведение матрицы в куб (алгоритм DD3) позволяет увеличить вероятность в десять раз.

По мере того как размер задачи увеличивается, разница в эффективность SRS и DDK становится всё более очевидной (см. рис. 3.8 и таблицу 3 для N = 512). Для N = 512 алгоритм SRS проигрывает по вероятности свыше двух порядков алгоритму DD3 (см. рис. 3.8).



**Рис. 3.8**. а) Плотность вероятности нахождения минимума. b) Отношение плотностей вероятности DD3 к SRS и DD2 к SRS, показанные на рис. а. (SK модель, N = 512)

**Таблица 3.2.** Сравнение эффективности алгоритма DDK с результатами стандартного алгоритма случайного поиска SRS для SK матриц размерности N = 100 и N = 512

	SRS	DD2	DD3	DD4	DD5
N=100					
$\delta E = \left(\overline{E}_m - E_0\right) / E_0$	10%	6.3%	3.4%	5.7%	2.8%
$\Pr[E_0, 0.99E_0]$	$2.7*10^{-2}$	9.3*10 <sup>-2</sup>	0.24	0.15	0.22
$P_0$	0.01	0.04	0.12	0.07	0.08
N=512					
$\delta E = \left(\overline{E}_m - E_0\right) / E_0$	9.4%	5.8%	3.5%	5.2%	3.0%
$\Pr[E_0, 0.99E_0]$	5*10 <sup>-5</sup>	$2*10^{-3}$	0.02	8.3*10 <sup>-3</sup>	0.037
$P_0$	<10 <sup>-7</sup>	4*10 <sup>-6</sup>	5*10 <sup>-5</sup>	$2.7*10^{-4}$	2*10 <sup>-3</sup>

Также интересно заметить, что величина  $\delta E$  не зависит существенно от N, т.е. средняя энергия минимумов находимых при случайном поиске отличается от энергии глобального на константу. При использовании SRS для SK матриц эта величина равна  $\delta E \approx 10\%$  с точностью до флуктуаций, убывающих как  $1/\sqrt{N}$ . Использование DD3 позволяет уменьшить величину  $\delta E$  почти втрое.

### 3.5в) Улучшение динамики. Алгоритм Хоудайера-Мартина.

Описанное выше повышение эффективности алгоритма обусловлены одной только статикой – трансформацией поверхности, по которой осуществляется спуск. Вместе с тем, можно модифицировать и саму динамику спуска. В ходе экспериментов было замечено, что очень часто спуск останавливается в мелких кавернах поверхности на расстоянии нескольких бит от глобального минимума. Для исключения этого нежелательного эффекта мы изменили алгоритм спуска

Мы взяли алгоритм локальной оптимизации из работы [1] (Houdayer and Martin (2001)). Этот алгоритм был инспирирован алгоритмом Кернигана-Лина [74] (Kernighan and Lin (1970)) и в работе [1] (Houdayer and Martin (2001)) адаптирован к задаче бинарной квадратичной минимизации. Этот алгоритм локального поиска аналогичен динамике Хопфилда. Однако он более эффективен благодаря способности выходить из мелких минимумов энергетической поверхности.

Здесь мы используем этот алгоритм в форме как он был предложен в работе [1] (Houdayer and Martin (2001)). Количество операций, выполняемых в процессе запуска одного старта динамики Хоудайера-Мартина примерно  $O_{HM} \approx 10O_{SRS}$  (см. таблицу 3.5). В силу более медленной работы этого алгоритма, в компьютерных экспериментах мы проводили лишь  $10^4$  стартов для данного типа динамики.

Использование новой динамики Хоудайера-Мартина вместо Хопфилдовой еще на порядок улучшило характеристики алгоритма DDK, которые приведены в Таблице 3.3 (SRS и DDK алгоритмы, базирующиеся на модифицированной динамике Хоудайера-Мартина мы будем обозначать аббревиатурами SRS-HM и DDK-HM соответственно).

На рис. 3.9 приведена плотность вероятности P(E), достигаемая модифицированными алгоритмами DDK-HM и SRS-HM. Сравнивая с кривыми рис. 3.6 видим, что указанная модификация динамики спуска привела к очень серьезному сдвигу спектральных кривых в сторону

глобального минимума. Как видим, вероятность попадания в глобальный минимум повысилась за счет модификации еще на порядок. Наилучший результат достигается алгоритмом DD3-HM: вероятность отыскания глобального минимума в  $2.5*10^4$  раз выше, чем для SRS (при N = 100).

**Таблица 3.3.** Эффективность алгоритмов SRS и DDK с динамикой Хоудайера-Мартина. Данные для двумерной модели Эдвардса-Андерсона при размерности *N* = 100, 196.

	SRS-HM	DD2-HM	DD3-HM	DD4-HM	DD5-HM
N=100					
$\delta E = \left(\overline{E}_m - E_0\right) / E_0$	4.7%	4.0%	2.6%	4.0%	2.5%
$\Pr[E_0, 0.99E_0]$	5.0%	8.2%	16.2%	9.0%	17.4%
$P_0$	1.4%	2.8%	6.0%	3.9%	5.9%
N=196					
$\delta E = \left(\overline{E}_m - E_0\right) / E_0$	5.9%	4.8%	3.4%	4.7%	3.2%
$\Pr[E_0, 0.99E_0]$	0.2%	0.5%	2.1%	0.6%	2.0%
$P_0$	4.7*10 <sup>-5</sup>	$2.2*10^{-4}$	4.3*10 <sup>-4</sup>	3.6*10 <sup>-4</sup>	$6.7*10^{-4}$



Рис. 3.9. Плотность вероятности P(E), получаемая модифицированным алгоритмом случайного поиска SRS-HM и DDK-HM при k = 2, 3, 4, 5. Для сравнения, пунктиром дана плотность вероятности, получаемая стандартным алгоритмом SRS. (данные для 2D EA модели с N = 100)

Для матриц Шеррингтона-Киркпатрика выигрыш от использования DDK-HM алгоритма вместо SRS-HM становится очевидным с увеличением размерности N. Результаты для SK матриц размера N = 512 показаны в таблице 3.4. Очевидно, что благодаря улучшенной динамике, SRS-HM алгоритм более эффективен нежели SRS (сравните с таблицей 3.2).

Таблица	3.4.	Эффективность	алгоритмов	SRS-HM	И	DDK-HM	c	динамикой
Хоудайер	a-Map	отина для SK матр	оиц при разме	ерности N	= 5	12.		

	SRS-HM	DD2-HM	DD3-HM	DD5-HM
$\delta E = \left(\overline{E}_m - E_0\right) / E_0$	4.5%	2.9%	1.7%	1.6%
$\Pr[E_0, 0.99E_0]$	0.015	0.08	0.22	0.23
<i>P</i> <sub>0</sub>	4.2*10 <sup>-4</sup>	$2.5*10^{-3}$	1.2*10 <sup>-2</sup>	9.5*10 <sup>-3</sup>

Как видно из таблицы 3.4 алгоритм DD3-HM немножко проигрывает алгоритму DD5-HM в средней энергии, однако выигрывает в вероятности отыскания глобального минимума.

алгоритм	Среднее время на 1000 стартов (с)
SRS	0.090
DD2	0.175
DD3	0.167
DD5	0.170
SRS-HM	0.950
DD2-HM	1.571
DD3-HM	1.211
DD5-HM	1.152

**Таблица 3.5.** Сравнение алгоритмов DDK по потреблению времени, SK N = 100

3.5г) Результаты для матриц 3-мерной модели Эдвардса-Андерсона (3D EA)

Интересно посмотреть работу предложенного алгоритма DDK и DDK-HM для задач, в которых глобальный минимум не известен и даже экстенсивный случайный поиск с огромным количеством стартов в этом не помогает.

Для этой цели мы выбрали матрицы модели 3-мерной модели Эдвардса-Андерсона, т.е. модель спинового стекла в трёхмерной кубической решётке с гауссовыми случайными взаимодействиями между соседними спинами. Было выбрано три размера:  $N = 8 \times 8 \times 8$ ,  $N = 10 \times 10 \times 10$  и  $N = 20 \times 20 \times 20$ . Как и раньше эксперименты включали в себя 10<sup>6</sup> стартов для динамики Хопфилда  $10^{4}$ динамики Хоудайера-Мартина. Результаты были и стартов для 100 усреднены по матрицам каждого размера. Ниже представлены результаты для алгоритмов DDK и DDK-HM только для наилучшего значения k = 3.



Рис. 3.10. а) Плотность вероятности P(E) нахождения минимумов с помощью алгоритма SRS (точечная кривая) и с помощью алгоритма DD3 (сплошная кривая); b) Сплошные кривые показывают плотность вероятности P(E), получаемую модифицированным алгоритмом случайного поиска DD3-HM; для сравнения, точечная кривая показывает плотность вероятности, получаемая стандартным алгоритмом SRS. Данные для 3D EA модели с L = 8, 10, 20. По оси абсцисс отложена энергия помноженная на  $\sqrt{N}$ .

На рис. 3.10а показана типичная картина распределения плотности вероятности для алгоритмов DD3 и SRS. Как видно из рисунка, кривые полученные для достаточно больших значений *N* (больше 500) хорошо

аппроксимируются гауссовым распределением с параметрами: 1) для DD3 среднее значение  $\overline{E_3} \approx -1.29 / \sqrt{N}$  и стандартное отклонение  $\sigma_3 \approx 0.41 / N$ ; 2) для SRS среднее  $\overline{E_1} \approx -1.16 / \sqrt{N}$  и стандартное отклонение  $\sigma_1 \approx 0.66 / N$ .

Алгоритм DD3 имеет существенное преимущество перед SRS. Более того, ни в одном из экспериментов с  $N \ge 1000$  алгоритм SRS не нашёл минимум с энергией  $E < \overline{E_3}$ . Этот результат вполне ожидаем поскольку  $(\overline{E_1} - \overline{E_3}) / \sigma_1 \approx 0.2\sqrt{N}$  и вероятность отыскания минимума с энергией  $E < \overline{E_3}$  оценивается формулой:

$$P(E < \overline{E_3}) \approx \frac{1}{0.2\sqrt{2\pi N}} e^{-0.02N}$$

которая даёт  $P < 1.3 \cdot 10^{-10}$  для N > 1000.

Результаты для DD3-HM показаны на рис. 3.10b. Пики описываются гауссовым распределением со средним и стандартным отклонением:

$$\overline{E} \approx -\frac{\varepsilon}{\sqrt{N}}, \quad \sigma \approx \frac{t}{N},$$

где  $\varepsilon = 1.331, 1.327, 1.315$  и t = 0.37, 0.40, 0.46для N = 512, 1000, 8000соответственно. Ясно что алгоритм DD3-HM показывает гораздо лучшие результаты чем DD3. Однако с ростом размерности это преимущество исчезает: пик кривой плотности вероятности сдвигается в сторону пунктирной линии, которая соответствует центру распределения DD3 на рис. 3.10a ( $\varepsilon \rightarrow 1.29$ ).

3.6 Применения алгоритма к задаче разбиения графа (graph bipartitioning)

Пусть имеется некоторый граф G, у которого N вершин  $V = (v_1, v_2, ..., v_N)$ , а веса рёбер описываются матрицей T размера  $N \times N$ . Без ограничения общности, считаем что T - симметричная матрица с нулевой диагональю ( $T_{ii} = 0$ ).

В данном разделе рассматривается задача разбиения графа: требуется разбить вершины графа на две равные непересекающиеся части  $V_1$  и  $V_2$  так, чтобы сумма весов рёбер, связывающих разные части, была минимальна. Задача сводится к минимизации суммы:

$$\min_{V_1, V_2} \sum_{i \in V_1} \sum_{j \in V_2} T_{ij}$$
(3.18)

при условии:

$$|V_1| = |V_2| = N / 2 \tag{3.19}$$

Вводя бинарные переменные  $s_i = \pm 1$  такие что,  $s_i = 1$  если  $i \in V_1$  и  $s_i = -1$  если  $i \in V_2$ , легко показать, что задача (3.18)-(3.19) сводится к минимизации квадратичного бинарного функционала:

$$E(S) = -\sum_{i=1}^{N} \sum_{j=1}^{N} T_{ij} s_i s_j, \ s_i = \pm 1$$
(3.20)

при ограничениях:

$$\sum_{i=1}^{N} s_i = 0 \tag{3.21}$$

Известно, что даже без ограничений, задача (3.21) представляет большой интерес, поскольку к ней сводится широкий класс *NP*-полных задач.

Для решения данной задачи разбиения графа известны такие эвристические методы, как: алгоритм Кернигана-Лина [74] (Kernighan and Lin (1970)), нейросетевая динамика Хопфилда [52] (Hopfield (1982)), симуляция отжига [39] (Kirkpatrick et al (1983)), спектральные методы [76] (Hagen and Kahng (1992)), различные многоуровневые схемы [75] (Karypis and Kumar (1997)), генетические алгоритмы [1] и мн. др.

Воспользуемся предложенным выше алгоритмом, который минимизирует функционал, построенный на матрице  $W = T^k$ :

$$E_k(S) = -\sum_{i=1}^N \sum_{j=1, i \neq j}^N W_{ij} s_i s_j, \ s_i = \pm 1$$
(3.22)

Данный подход развивается в этом параграфе для задачи разбиения графа.

*3.6а) Методика*. В качестве алгоритма разбиения графа был выбран простейший алгоритм случайного поиска, суть которого в следующем:

i) Случайным образом генерируется начальное разбиение графа на две равные части. ii) Дальше, ищется пара вершин из разных частей, перестановка которых понижает значение функционала (3.20). iii) До тех пор пока такие вершины есть, то они меняются местами и ищется следующая пара. iiii) Если же такой пары нет, то алгоритм останавливается.

Initial partition  $S = (s_1, s_2, ..., s_N), s_i = \pm 1$  such that  $\sum_{i=1}^N s_i = 0$ For each i from 1 to N $h_i = \sum_{i=1, j=1}^N T_{ij} s_j$ endfor stop = 1
While (stop > 0) stop = 0**For** each i from  $V_1$ For each j from  $V_2$ **If** (  $h_i s_i + h_j s_j - 2T_{ij} s_j s_j < 0$ )  $s_i = -s_i$  $s_j = -s_j$ For each k from 1 to N $h_k = h_k + 2T_{ki}s_i + 2T_{ki}s_i$ Remove *i* from  $V_1$  and add it to  $V_2$ Remove j from  $V_2$  and add it to  $V_1$ stop = 1endif endfor endfor endwhile  $E = -\sum_{i=1}^{N} h_i s_i$ 

Рис. 3.11. Алгоритм локальной оптимизации, используемый при случайном поиске в задаче разбиения графа.

Эта процедура повторяется много раз, до тех пор, пока не найдётся разбиение, удовлетворяющее по величине разреза, либо не выйдет время на поиск. Алгоритм оптимизации представлен на рис.3.11. Количество операций алгоритма  $O(N^2)$ .

В дальнейшем будем рассматривать только случай, когда матричные элементы *T<sub>ij</sub>* случайны и могут быть как положительны, так и отрицательны. Для этого случая, как было показано выше:

1. Функционал (3.20) в общем случае имеет множество локальных минимумов в бинарном пространстве переменных  $s_i = \pm 1$ . Минимумы имеют

различные радиусы области притяжения и, следовательно, различную вероятность отыскания при случайном поиске.

2. Радиус области притяжения минимума пропорционален глубине минимума и, следовательно, вероятность отыскания минимума экспоненциально возрастает с ростом глубины минимума. Иными словами, в процессе случайного поиска с наибольшей вероятностью находятся наиболее глубокие минимумы (наиболее хорошие решения).

3. Можно трансформировать энергетическую поверхность функционала (3.20) так, чтобы его глубокие минимумы стали ещё глубже и значительно шире, а радиусы аттракторов мелких минимумов уменьшились бы. Такую трансформацию можно получить путём возведения матрицы T в степень k = 2, 3, ...

4. При возведении матрицы в степень, минимумы не только меняют глубину, но и сдвигаются. Однако наиболее глубокие минимумы функционала с матрицей  $W = T^k$  расположены вблизи от глубоких минимумов исходного функционала (3.22).

5. Чем больше размерность задачи *N*, тем больше относительное увеличение области притяжения глубоких минимумов при возведении матрицы в степень.

Все эти положения были получены для задачи минимизации квадратичного бинарного функционала (3.20) без ограничений вида (3.21). Однако, вероятно, что эти же правила будут справедливы и для задачи с ограничениями, типа задачи о разбиении графа. Так, например, из практики известно, что при случайном поиске хорошие решения (с малой величиной разреза) попадаются чаще, чем плохие решения. А возведение матрицы в степень может значительно повысить эту вероятность.

Таким образом, предлагаемый метод состоит из следующих двух шагов:

**1-ый шаг.** Создаём граф  $G_k$  с матрицей связей  $T^k$  (k = 2, 3, ..., 5) с нулевой диагональю. Генерируем случайное начальное разбиение на две равные части и, используя алгоритм локальной оптимизации, описанный на

рис. 3.11, находим субоптимальное (или оптимальное) разбиение  $S_k$  графа  $G_k$ .

**2-ой шаг.** Полученное решение *S<sub>k</sub>* даёт нам *начальное разбиение* для поиска оптимального разбиения исходного графа *G* с матрицей связи *T*. Снова применяем алгоритм локальной оптимизации и находим субоптимальное (или оптимальное) разбиение *S* графа *G*.

Если величина разреза полученного решения недостаточно мала, то описанная 2-х этапная процедура повторяется.

3.66) Результаты. Для проверки эффективности предлагаемого алгоритма был проведен ряд компьютерных экспериментов. В качестве тестовых были взяты графы тех же двух типов, что и раньше:

Первый тип — это графы двумерной модели Изинга. Граф этого типа представляет собой двумерную квадратную решётку, где связи есть только между ближайшими соседями, величина связи случайная, распределена по нормальному закону относительно нуля. Такие графы сильно разрежены, т.е. число рёбер много меньше чем  $N^2$ .

Для второго типа графов – графы модели Шеррингтона-Киркпатрика. Матрица связей графа в этом случае полносвязная (число рёбер  $\Box N^2$ ), элементы случайны и имеют равномерное распределение на некотором отрезке, симметричном относительно нуля.

Тесты проводились при размерностях  $N \approx 100, 200$ . Относительно небольшая размерность задачи выбрана для того, чтобы была возможность найти глобальный минимум и оценить эффективность алгоритма. В качестве «глобального» минимума выбирался самый глубокий из найденных за все проведённые эксперименты минимум. Фактически, оказалось, что от эксперимента к эксперименту наблюдается одно и то же значение самого глубокого минимума, с достаточно хорошей повторяемостью. Поэтому в дальнейшем не будем различать найденный «глобальный» минимум и истинный глобальный минимум функционала.

Сперва оценим работу алгоритма, приведенного на рис. 3.11, без возведения матрицы в степень. В экспериментах после  $10^5$  стартов была посчитана средняя энергия минимумов и её разница с энергией глобального минимума. Эксперименты проводились для набора из 50 матриц каждого типа. Одним из параметров, характеризующих эффективность алгоритма минимизации, была величина  $\delta E = (E_0 - \overline{E}_m) / E_0$ , где  $E_0$  – энергия глобального минимума, а  $\overline{E}_m$  – средняя энергия находимых минимумов. Относительная разница  $\delta E$  является довольно-таки постоянной величиной, самоусредняющейся с ростом размерности N. Например, для полносвязных матриц с равномерным распределением элементов стандартный алгоритм

 $\delta E = 8.0\% \pm 1.0\%$  при N = 100

 $\delta E = 8.5\% \pm 0.8\%$  при N = 200

а в случае матриц Изинга имеем:

 $\delta E = 10.4\% \pm 1.1\%$  при N = 100 $\delta E = 10.5\% \pm 0.9\%$  при N = 196

Другим параметром, характеризующим эффективность алгоритма минимизации, является величина:

 $P = \Pr\{E \in [E_0, 0.99E_0]\},\$ 

т.е вероятность отыскания очень глубоких минимумов с энергией  $E \in [E_0, 0.99E_0]$  в узком интервале вблизи энергии глобального минимума.

В дальнейших экспериментах строился граф с матрицей связей  $T^k$  для k = 2, 3..., 5. Разбиение этого графа использовалось как начальное разбиение при минимизации разреза исходного графа с матрицей T. Т.е. запускался двухэтапный алгоритм, наподобие DDK. Полученные результаты приведены в таблице 3.6 и на рис. 3.12.

Из графиков и таблицы видно, что с возведением матрицы в степень разница между средней энергией минимумов и энергией глобального минимума уменьшается. Как следствие, повышается вероятность попадания в более глубокие минимумы: для полносвязных графов размера *N* = 100 повышение вероятности в 3-6 раз, для графов двумерной модели Изинга в 10-30 раз. Для больших размерностей, повышение вероятности на порядки.

**Таблица 3.6.** Полученные результаты для возведения матрицы в степень. Два типа матриц – полносвязные матрицы модели SK (левая половина таблицы) и матрицы 2-мерной решётки Изинга (правая половина), для двух размерностей – N = 100 (верхняя половина) и  $N \approx 200$  (нижняя половина).

	Матрицы м	одели SK N = 100	Матрицы 2D	Изинга N = 100	
	$\delta E$	$\Pr\{E \in [E_0, 0.99E_0]\}$	$\delta E$	$\Pr\{E \in [E_0, 0.99E_0]\}$	
Т	8.0% ± 1.0%	0.05	$10.4\% \pm 1.1\%$	0.001	
$T^2$	$5.0\% \pm 1.3\%$	0.16	8.0% ± 1.3%	0.011	
$T^3$	$2.8\% \pm 1.0\%$	0.31	$5.4\% \pm 0.9\%$	0.034	
$T^4$	$4.3\% \pm 2.0\%$	0.23	$7.6\% \pm 1.3\%$	0.018	
$T^5$	$2.4\% \pm 1.3\%$	0.32	$4.5\% \pm 0.9\%$	0.052	
	Матрицы мо	одели SK <i>N</i> = 200	Матрицы 2D Изинга <i>N</i> = 196		
	$\delta E$	$\Pr\{E \in [E_0, 0.99E_0]\}$	$\delta E$	$\Pr\{E \in [E_0, 0.99E_0]\}$	
Т	$8.5\% \pm 0.8\%$	0.01	10.5%± 0.9%	1.3*10 <sup>-5</sup>	
$T^2$	$5.0\% \pm 0.9\%$	0.08	8.4% ± 1.0%	4.8*10 <sup>-4</sup>	
$T^3$	$2.8\%\pm0.7\%$	0.19	$5.1\% \pm 0.7\%$	6.2*10 <sup>-3</sup>	
$T^4$	4.5% ± 1.5%	0.12	$8.0\% \pm 1.0\%$	$1.2*10^{-3}$	
$T^5$	$2.5\% \pm 0.9\%$	0.21	$4.2\% \pm 0.7\%$	$1.7^{*}10^{-2}$	



**Рис. 3.12**. Сдвиг в распределении плотности вероятности при возведении матрицы в куб. Штрихпунктирными линиями показаны распределения плотности вероятности для стандартного алгоритма (без возведения матрицы в степень), сплошными - для алгоритма с возведением матрицы в куб. Энергия E = -1 соответствует энергии глобального минимума. Данные получены для матриц Изинга размерности N = 100 и 196.

## 3.7 Выводы по главе 3

В главе 3 мы описали двухэтапный алгоритм минимизации DDK.

Подготовительная фаза состоит в следующем: 1) исходная матрица T симметризуется (если она изначально не симметрична) и диагональные матричные элементы обнуляются; 2) матрица возводится в k-ю степень и в полученной матрице  $M = T^k$  обнуляются диагональные элементы; 3) на основе матрицы M строится квадратичный функционал  $E_k(S)$ .

После подготовки можно приступать к процедуре случайного поиска на основе алгоритма двухэтапного спуска, состоящего в следующем: *a*) на первом этапе из начальной конфигурации осуществляется спуск по поверхности  $E_k(S)$  в ближайший локальный минимум  $S_m^{(k)}$  функционала  $E_k(S)$ ; *b*) на втором этапе проводится коррекция – из точки  $S_m^{(k)}$  по поверхности  $E_1(S)$  спускаемся в ближайший локальный минимум  $S_m$ 

Сравнение показывает, что трансформация поверхности (основанная на возведении матрицы T в степень) приводит к существенному повышению эффективности оптимизационного алгоритма. В частности, для матриц Эдвардса-Андерсона размера N = 196, использование алгоритма DD3-HM позволяет добиться следующих улучшений: вероятность попадания в глобальный минимум увеличивается на порядок величины по сравнению с алгоритмом SRS-HM; разница между средней энергией  $\overline{E}_m$  и энергией глобального минимума уменьшается вдвое; вероятность попадания в интервал энергий близких к глобальному минимуму  $E_m \in [E_0, 0.99E_0]$ вырастает на порядок величины.

Анализ показывает, что алгоритм DD3 является оптимальным. Средняя эффективность алгоритма DD5 близка и даже лучше чем у DD3, однако DD5 менее стабилен. Например, для 2D EA матриц размера N = 100 алгоритм DD5 не находит глобальный минимум для 60% примеров, т.е. его

эффективность очень высока для примерно 40% задач, и очень плоха для оставшихся 60%.

Добавление динамики Хоудайера-Мартина вместо динамики Хопфилда также значительно улучшает алгоритм. В частности, алгоритм DD3-HM позволяет найти глобальный минимум с вероятностью примерно 6%, которая в 30 раз выше чем вероятность (порядка 0.2%) даваемая алгоритмом DD3 и в 25000 раз больше чем вероятность ( $\Box 2.4 \cdot 10^{-6}$ ) даваемая алгоритмом SRS для 2D EA матриц размера N = 100.

Введение динамики Хоудайера-Мартина увеличивает время одного спуска примерно в десять раз (см. таблицу 3.5). Однако эта динамика добавляет стабильность в алгоритм: не было ни одной матрицы 2D EA размера N = 100, для которой DD3-HM алгоритм не нашёл бы глобальный минимум.

Итоговый вывод следующий: во всех экспериментах алгоритм двойного спуска (DDK или DDK-HM) с очень большой вероятностью приводил систему в конфигурацию, очень близкую к глобальному минимуму, как по энергии, так и по расстоянию. При этом все характеристики алгоритма значительно лучше по сравнению с алгоритмом SRS. Кроме того, преимущество алгоритма DDK (DDK-HM) над алгоритмом SRS только растёт с ростом размерности N. Исследование экспериментальных данных показывает, что DD3-HM является наилучшим выбором для практического использования: ОН проявляет наибольшую вероятность отыскания глобального минимума, наибольшую стабильность и наименьшее время работы.

Также в данной главе мы описали применение алгоритма DDK к задаче минимизации разреза графа на две равные части. На первом шаге предлагается делать разбиение графа с матрицей связи  $T^k$ , где T - матрица связи исходного графа, k - некоторая степень (k = 2, 3, ..., 5). Полученное на первом шаге разбиение используется как начальное на втором шаге. Как

показывают эксперименты, такой двухступенчатый алгоритм существенно повышает эффективность алгоритма разбиения графа.

А именно, эксперименты показывают, что, во-первых, такая трансформация связей графа понижает среднюю энергию находимых минимумов (разница между энергией глобального минимума и средней энергией находимых минимумов уменьшается почти в 3 раза, рис. 3.12). Как следствие, вероятность отыскания близких к глобальному минимумов повышается на порядки (для матриц Изинга уже при N = 100, а также для матриц с равномерным распределением при больших размерностях начиная с N = 200).

Более того, с ростом размерности задачи, эффективность от использования возведения матрицы в степень, стремительно растёт.

Все эти результаты справедливы, какой бы ни был использован алгоритм локальной оптимизации разбиения графа: простейший жадный алгоритм (использованный в данной работе), алгоритм Кернигана-Лина, алгоритм имитации отжига или др.

В данной работе мы рассмотрели два типа графов, в которых матрицы связей имеют как положительные, так и отрицательные элементы. В практических задачах часто возникает потребность разбиения графов, рёбра которых неотрицательны. Дальнейшие исследования могут быть связаны с анализом эффективности предложенного алгоритма для такого рода матриц. Однако нами данный вопрос не исследовался.

### 4. АЛГОРИТМ MIX-MATRIX (MM)

В этом разделе мы опишем ещё один алгоритм минимизации, главным элементом которого, как и в алгоритме DDK, описанном в главе 3, является трансформация энергетической поверхности функционала, направленная на то, чтобы глубокие минимумы исходного функционала стали ещё глубже, и приводящая тем самым к экспоненциальному увеличению вероятности нахождения глубоких минимумов при случайном поиске.

Трансформировать описываемую квадратичной формой E(S) поверхность можно только трансформируя матрицу, на которой она построена. Заменим матрицу исходного функционала (1.1) на новую матрицу, которую мы назовём Микс-Матрицей:

$$M = \frac{1}{\sigma_T} (1 - z)T + \frac{1}{\sigma_{2T}} zT^2 , \qquad (4.1)$$

где  $T^2$  - матрица, получаемая возведением матрицы T в квадрат с последующим обнулением диагональных элементов,  $\sigma_T$  и  $\sigma_{2T}$  среднеквадратичное отклонение элементов матриц T и  $T^2$  соответственно. При изменении параметра z от 0 до 1 происходит переход от матрицы M = Tк матрице  $M = T^2$ . Соответственно, поверхность, описываемая функционалом E(S), трансформируется в поверхность, описываемую функционалом  $E_z(S)$ :

$$E_{z}(S) = -\frac{1}{\sigma_{M}N^{2}} \sum_{i=1}^{N} \sum_{j=1}^{N} M_{ij} s_{i} s_{j}$$
(4.2)

где  $\sigma_M = \sigma_T \sqrt{z^2 + (1-z)^2}$  - стандартное отклонение элементов матрицы M, диагональные элементы матрицы M строго равны нулю  $(M_{ii} = 0)$ . Минимизация функционала (4.2) производится так же, как в стандартном алгоритме случайного поиска. Вычисляется величина действующего на произвольно выбранный *i*-й спин  $h_i^{(z)} = \sum M_{ij} s_j$  и если спин находится в неустойчивом состоянии  $(h_i^{(z)} s_i < 0)$ , то его состояние обновляется в соответствии с решающим правилом  $s_i = \operatorname{sgn} h_i^{(z)}$ . Эта процедура последовательно применяется ко всем спинам до тех пор, пока сеть не конвергирует в устойчивое состояние, соответствующее минимуму функционала  $E_z(S)$ .

Ниже мы покажем, что при такой трансформации функционала, глобальный минимум и другие глубокие минимумы исходного функционала станут глубже. Разумеется, побочным эффектом такой трансформации является то, что минимумы сдвигаются в пространстве. Однако будет показано, что сдвиг глобального минимума невелик.

Прежде чем перейти к последующему анализу напомним, что, как показано в главе 2, форма минимума задается двумя параметрами: средним значением квазиэнергии  $\lambda = h_i s_i / \sqrt{N}$  (т.е. глубиной минимума) и ее дисперсией  $\sigma_{\lambda}^2$ . Соответственно, все последующие выражения будут содержать эти два независимых параметра. В произвольной точке вдали от минимума величина  $\lambda$  может принимать как положительные, так и отрицательные значения, и ее дисперсия максимальна  $\sigma_{\lambda}^2 = 1$ . В минимуме величина  $\lambda$  может быть только положительной ( $h_i s_i > 0$ , i = 1, N) и величина  $\sigma_{\lambda}^2$  уменьшается до значений  $\sigma_{\lambda}^2 \approx 0.5$ .

# 4.1 Углубление минимума

Покажем что трансформация (4.1) приводит к углублению минимума. Рассмотрим энергию  $E_{z0} = E_{z0}(S_0)$  в точке  $S_0$ . Из (4.2) после ряда преобразований получаем:

$$E_{z0} = E_0 \frac{1 - z + zA}{\sqrt{(1 - z)^2 + z^2}}, \qquad A = \frac{\sigma_\lambda^2 + \gamma^2 - 1}{\gamma} \sqrt{q/(q - 1)} \quad , \tag{4.3}$$

где q – число отличных от нуля элементов в строке матрицы T (q = N - 1 в SK-модели и q = 4 в модели 2D EA). Аналогичные вычисления показывают, что дисперсия углубления  $\sigma^2(E_{z0}/E_0)$  стремится к нулю с ростом

размерности как 1/N. Нетрудно заметить, что углубление имеет место только для достаточно глубоких минимумов, удовлетворяющих соотношению  $\gamma^2 > 1 - \sigma_{\lambda}^2$ . Мелкие минимумы ( $\gamma^2 < 1 - \sigma_{\lambda}^2$ ) в результате трансформации становятся еще более мелкими.

Из (4.3) следует, что имеется оптимальное значение параметра z

$$z_{opt} = \frac{A}{1+A},\tag{4.4}$$

при котором углубление достигает максимального значения:

$$(E_{z0} / E_0)_{opt} = \sqrt{1 + A^2}$$
(4.5)

Для матриц 2D EA имеем  $\gamma \approx 1.31$ ,  $\sigma_{\lambda} \approx 0.69$  и из (4.4)-(4.5) находим  $z_{opt} \approx 0.51$ ,  $(E_{z0} / E_0)_{opt} \approx 1.45$ . Аналогично, для SK-модели  $\gamma \approx 1.51$ ,  $\sigma_{\lambda} \approx 0.76$ , значит  $z_{opt} \approx 0.55$ ,  $(E_{z0} / E_0)_{opt} \approx 1.59$ . Видим, что оптимальное значение z находится где-то вблизи 0.5.

Эксперимент подтверждает полученные соотношения. На рис. 4.1 показана характерная картина углубления на примере глобального минимума. Как видим, трансформация приводит к увеличению глубины: почти на 60% для полносвязных матриц SK-модели при  $z \approx 0.55$  и примерно на 45% для разреженных матриц модели 2D EA при  $z \approx 0.51$ .



**Рис. 4.1** Увеличение глубины глобальных минимумов в зависимости от параметра z. Маркеры с планками - экспериментальные кривые: SK 500 - для N = 500 модели Шеррингтона-Киркпатрика; 2D EA 484 - для матриц 2D модели Эдвардса-Андерсона размерности  $N = 22 \times 22$ . Штриховые линии – теоретические, построенные по формуле (4.3) с параметрами  $\gamma \approx 1.51$ ,  $\sigma_{\lambda} \approx 0.76$  для SK 500, и  $\gamma \approx 1.31$ ,  $\sigma_{\lambda} \approx 0.69$  для 2D EA 484.

### 4.2 Сдвиг минимума

Оценим насколько сдвинется минимум при трансформации поверхности. Среднюю величину сдвига можно представить в виде

$$\overline{d} = N \cdot P, \qquad (4.6)$$

где  $P = \Pr\left\{h_i^{(z)}s_i^{(0)} < 0 | h_i s_i^{(0)} > 0\right\}$  - вероятность того, что направления спина  $s_i$ и локального поля  $h_i^{(z)}$  не совпадают для микс-матрицы, при условии, что они совпадают при исходной матрице. Для искомой вероятности (см. Приложение A) получаем:

$$P = \frac{1}{\sqrt{2\pi\sigma_{\lambda}\Phi(\gamma/\sigma_{\lambda})}} \int_{0}^{\infty} dx \, e^{-\frac{(x-\gamma)^{2}}{2\sigma_{\lambda}^{2}}} \left[ 1 - \Phi\left(\frac{\alpha x + \sigma_{\lambda}^{2} - 1}{\sigma_{\lambda}^{2}}\right) \right], \tag{4.7}$$

где  $\Phi(\cdot)$  - интеграл вероятностей,  $\alpha = \left[ (1-z)\sqrt{(q-1)/q} + z\gamma \right]/z$ . Отметим, что при  $z \to 0$  интеграл в (4.7) стремится к нулю, т.е. сдвига нет, поскольку функционал  $E_{z=0}(S)$  совпадает с исходным E(S). Еще более важно отметить, что наименьшие сдвиги ожидаются для наиболее глубоких минимумов.

Как и следовало ожидать, формула (4.7) описывает монотонное увеличение величины  $\overline{d}$  с ростом z. Это подтверждается экспериментом (см. рис. 4.2). Более того, с ростом z возрастают и флуктуации величины сдвига d. Это ожидаемый результат, поскольку аналогичные (4.6)-(4.7) вычисления показывают, что дисперсия сдвига описывается выражением  $\sigma^2(d) = \overline{d}$ .



**Рис. 4.2.** Экспериментально полученный сдвиг минимума как функция от z для двух типов матриц: SK 500 - для N = 500 модели Шеррингтона-Киркпатрика; 2D EA 484 - для матриц 2D модели Эдвардса-Андерсона размерности  $N = 22 \times 22$ .

Как показывает эксперимент (см. рис. 4.2), максимальный сдвиг глобального минимума, достигаемый при z = 1, не превышает величины  $\overline{d}_{max} = 0.06N$  для полносвязных матриц SK-модели и величины  $\overline{d}_{max} = 0.03N$  для разреженных матриц модели 2D EA. Оптимальные значения, достигаемые при  $z = z_{opt}$ , приблизительно в 3 раза меньше:  $\overline{d}_{opt} = 0.015N$  для SK-модели и  $\overline{d}_{opt} = 0.01N$  для модели 2D EA.

# 4.3 Описание алгоритма ММ

Как было показано выше, трансформация функционала приводит к существенному (до 45-60%) увеличению глубины глобального минимума. Тем самым, вероятность отыскания минимума экспоненциально по N увеличивается. Однако, при этом происходит незначительный сдвиг минимума в пространстве на расстояние  $\overline{d} \sim 0.01N - 0.02N$ . Чтобы скорректировать этот сдвиг, мы предлагаем следующий алгоритм минимизации, состоящий из двух шагов:

1) На первом шаге из случайной точки выполняется спуск по поверхности  $E_z(S)$  и находится конфигурация  $S_m^{(z)}$ , соответствующая минимуму функционала  $E_z(S)$ .

2) Второй шаг включает коррекцию: из точки  $S_m^{(z)}$ , мы спускаемся по поверхности E(S) до ближайшего минимума  $S_m$  функционала E(S).

## 4.4 Результаты для алгоритма ММ

Для проверки эффективности предложенного алгоритма в процессе численных экспериментов строилась Микс-Матрица при значениях z от 0 до 1 с интервалом  $\Delta z = 0.05$ . Результаты экспериментов усреднялись по ансамблю из 50 матриц. Каждый эксперимент включал в себя по  $N_{runs} = 10^6$ стартов. Каждый старт приводил к некоторому локальному минимуму. Мы следили за средней энергией  $\overline{E}$  найденных минимумов, за вероятностью попадания в интервал энергий  $E \in [-1; -0.99]$ , близких к глобальному (за -1 принята энергия глобального минимума  $E_0$ ), а также за вероятностью отыскания глобального минимума (при малых размерностях  $N \le 500$ ). В экспериментах помимо описанной выше микс-матрицы (4.1) мы также попробовали использовать микс-матрицу типа  $M = aT + bT^3$ .

Типы матрии. Эффективность предложенного двухэтапного 4.4a) проверена для разных ВИДОВ случайных алгоритма была матриц: SK-модели (модель Шеррингтона-Киркпатрика) и полносвязных матриц разреженных матриц моделей 2D EA и 3D EA(модель Эдвардса-Андерсона) с периодическими граничными условиями. Отыскание глобального минимума функционала энергии E(S) для данных типов матриц является NP-сложной задачей (исключение составляет 2D EA - модель, для которой этот вопрос ещё окончательно не решён). Как следствие, нахождение основного (глобального минимума) небольших состояния возможно ЛИШЬ для размерностей. Для нахождения основных состояний SK матриц, мы решали задачу разными алгоритмами (SRS, алгоритм DDK (глава 3), GRA-алгоритм [1]), проводили большое количество стартов и выбирали самые глубокие найденные минимумы в качестве «основных» состояний. Пользуясь таким подходом, с большой степенью уверенности можно сказать, что нам удалось найти основные состояния для матриц размерности вплоть до N = 500. Для 2D и 3D EA матриц основные состояния были взяты с сайта сервера спинового стекла1, который находит их с помощью branch and cut метода [33].

<sup>1</sup> Spin Glass Server. URL http://www.informatik.uni-koeln.de/spinglass/



**Рис. 4.3.** Среднее значение энергии локальных минимумов, найденных с помощью алгоритма ММ. Пунктирные линии – mix-матрица с  $T^3$ , сплошные - mix-матрица с  $T^2$ . Два типа матриц – SK и 2D EA. По оси Y величина средней энергии  $\overline{E}$  поделена на энергию глобального минимума  $E_0$  и примерно одинакова для разных N.

4.46) Средняя энергия минимумов. На рис. 4.3 показано как меняется средняя энергия  $\overline{E}$  найденных минимумов при различных z. Интересно отметить, что величина  $\overline{E} / E_0$  не зависит от размерности задачи, а зависит лишь от типа матрицы. Причём, результаты для 3D Изинга сливаются с результатами для 2D Изинга и потому не приведены на графике.

Как видно из рис. 4.3, с ростом z средняя энергия  $\overline{E}$  приближается к  $E_0$ . Для mix-матрицы с примесью  $T^2$  наблюдается максимум при  $z \approx 0.7$ , хотя максимальное углубление минимумов (см. пункт 4.3) достигается при z = 0.51 - 0.55. Для микс-матрицы с  $T^3$  рост средней энергии  $\overline{E}$  монотонный вплоть до z = 1. Более того, можно продолжить увеличивать z дальше за 1 и обнаружить что пик  $\overline{E}$  приходится примерно на z = 1.2. Впрочем, величина  $\overline{E}$  при z = 1.2 всего лишь на доли процента отличается от значения  $\overline{E}$  при z = 1 (не показано).

Самым важным является то, что независимо от размерности задачи алгоритм MM позволяет находить минимумы, всего лишь на несколько процентов отличающиеся от энергии глобального минимума. Так, для матриц двумерной и трёхмерной модели EA алгоритм SRS (z=0) даёт минимумы с энергией на 17% хуже глобального, в то время как алгоритм MM (при z = 0.7) даёт минимумы отличающиеся от энергии глобального минимума лишь на 6.5 - 7% (см. табл. 4.1). Аналогично, для матриц SK модели, алгоритм SRS не доходит до глобального минимума примерно на 10%, в то же время средняя глубина минимумов, получаемых алгоритмом MM при z = 0.7, оказывается лишь на 4% меньше глубины глобального минимума (см. табл. 4.1).



Рис. 4.4. Логарифм отношения вероятностей попадания в интервал энергий [-1, -0.99]. Пунктирные линии – mix-матрица с  $T^3$ , сплошные - mix-матрица с  $T^2$ . Слева: результат для матриц SK размерности N = 500 ( $P_{SRS} \approx 3.10^{-5}$ ). Справа: результат для матриц 2D EA размерности N = 144 ( $P_{SRS} \approx 2.6.10^{-7}$ ), при малых z на некоторых матрицах алгоритм не находил глобальных минимум., поэтому точки отсуствуют.

4.4в) Вероятность попадания в интервал [-1;-0.99]. На рис. 4.4 показано во сколько раз увеличивается вероятность нахождения минимумов с энергией отличающейся от глобальной меньше чем на 1%. Для демонстрации мы выбрали максимально возможные размерности задач, с которыми мы ещё можем справиться с помощью обычного случайного поиска. Для матриц 2D EA уже при размерности  $N = 12 \times 12$  вероятность отыскания алгоритмом SRS минимумов с энергией  $E \in [-1; -0.99]$  не больше чем  $P_{SRS} = 3 \cdot 10^{-7}$ . Для SK матриц максимальная размерность N = 500(вероятность  $P_{SRS} = 3 \cdot 10^{-5}$ ). Вероятность получаемую при помощи алгоритма MM мы обозначили как  $P_{MM}$ . Как видно из рис. 4.4, разница между  $P_{MM}$  и *P*<sub>*srs*</sub> даже при относительно небольшой размерности задачи составляет примерно 3 порядка.

Интересно, что для SK матриц ход кривых на рис. 4.4 для mix-матриц с  $T^2$  и  $T^3$  практически совпадает, и лишь при z > 0.7 кривые начинают расходиться. Для матриц 2D EA имеет место другая картина: mix-матрица с  $T^2$  имеет явное преобладание вплоть до  $z \approx 0.8$ , после чего начинает преобладать mix-матрица с  $T^3$ .

Видно также (рис. 4.4), что с ростом z начинает увеличиваться дисперсия, показанная на рисунке планками стандартного отклонения, что потенциально может привести к нестабильности алгоритма, т.е. для некоторых задач трансформация функционала может привести не к улучшению поиска, а наоборот, к ухудшению. С ростом размерности Nдисперсия должна спадать, а отношение  $P_{MM}$  к  $P_{SRS}$  резко увеличиваться. Однако это трудно продемонстрировать из-за того, что величины  $P_{MM}$  и  $P_{SRS}$ с ростом N уменьшаются экспоненциально и нет возможности их экспериментальной оценки.

4.4г) Предел больших размерностей. Интересно посмотреть работу предложенного алгоритма для матриц, для которых мы не знаем глобального минимума и даже экстенсивный случайный поиск с огромным числом стартов нам в этом не помогает. Для этой цели мы выбрали матрицы 3-х мерной модели Эдвардса-Андерсона. Были взяты четыре размерности  $N = L \times L \times L$  при L = 5, 8, 10 и 20. В эксперименте определялась плотность вероятности  $P(\gamma)$ , где  $P(\gamma)d\gamma$  есть вероятность отыскания минимума с глубиной в интервале  $[\gamma, \gamma + d\gamma]$ . На рис. 4.5 показаны графики плотности вероятности, полученные SRS-алгоритмом и MM-алгоритмом (при z = 0.7). При достаточно больших N полученные кривые хорошо аппроксимируется гауссианом

$$P(\gamma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{\left(\gamma - \overline{\gamma}\right)^2}{2\sigma^2}\right]$$
(4.8)

с параметрами  $\gamma$  и  $\sigma$ , показанными в таблице 4.1.

Как видим, вследствие трансформации спектр минимумов сдвигается в сторону глобального минимума в среднем на  $\Delta \gamma = 0.14$ . Это весьма существенный сдвиг: расстояние от пика до глобального минимума уменьшается более чем в 2.5 раза. Наиболее отчетливо это заметно при больших размерностях, поскольку с ростом размерности ширины пиков уменьшаются как  $1/\sqrt{N}$ . Для сравнения отметим, что уже при  $N \ge 10^3$  ни в одном из экспериментов алгоритм SRS не мог найти минимум хотя бы той же глубины, что и средняя глубина минимумов  $\gamma_{z=0.7} \approx 1.29$ , находимых с помощью MM алгоритма. Это легко понять, поскольку для алгоритма SRS вероятность отыскать минимум с глубиной  $\gamma > \gamma_{z=0.7}$  можно оценить с помощью (4.8) в виде:

$$\operatorname{Pr}_{SRS}(\gamma \ge \overline{\gamma}_{z=0.7}) \approx \frac{1}{\sqrt{2\pi N}} e^{-0.023N}$$
(4.9)

что при  $N \ge 10^3$  дает значение  $\Pr_{SRS} < 10^{-12}$ . Фактически это означает, что при  $N > 10^3$  алгоритм ММ по скорости отыскания достаточно глубоких минимумов ( $\gamma > 0.96\gamma_0$ ) более чем на 12 порядков превосходит стандартный алгоритм SRS.



Рис. 4.5. Плотности распределения минимумов, полученных алгоритмом SRS (z = 0) – штриховые линии, и с помощью MM алгоритма (z = 0.7) - сплошные линии. Результаты для 3D EA модели размерности  $N = L^3$  для L = 5,8,10,20. По оси абсцисс отложена величина  $\gamma \equiv |E|\sqrt{N}$ , поскольку при такой нормировке энергии положение пиков не зависит от размерности.

В таблице 4.1 приведены значения параметров, описывающих спектральное распределение (4.8), для 3-х типов матриц: 2D EA, 3D EA и SK, а также приведены асимптотические формулы для глубины глобального минимума  $\gamma_0$ , средней глубины  $\overline{\gamma}$  и полуширины спектра  $\sigma$  находимых минимумов при больших N.

**Таблица 4.1**. Сводная таблица значений глубины глобального минимума  $\gamma_0$ , средней глубины  $\overline{\gamma}$  и полуширины спектра  $\sigma$ , получаемых алгоритмами SRS и MM.

		SRS $(z=0)$		MM (z = 0)	0.7)
Ν	${\gamma}_0$	$\overline{\gamma}$	σ	$\overline{\gamma}$	$\sigma$
10x10	$1.316 \pm 0.031$	$1.099 \pm 0.023$	0.058	$1.234 \pm 0.028$	0.037
12x12	$1.319 \pm 0.025$	$1.096 \pm 0.021$	0.049	$1.234 \pm 0.022$	0.033
14x14	$1.313 \pm 0.023$	$1.098 \pm 0.017$	0.041	$1.230 \pm 0.019$	0.026
16x16	$1.313 \pm 0.021$	$1.098 \pm 0.013$	0.035	$1.229 \pm 0.018$	0.023
22x22	$1.314 \pm 0.009$	$1.098 \pm 0.010$	0.026	$1.227 \pm 0.009$	0.017
32x32	$1.314 \pm 0.009$	$1.099 \pm 0.008$	0.018	$1.229 \pm 0.007$	0.012
$N \rightarrow \infty$	$1.31 \pm 0.3 / \sqrt{N}$	$1.09 \pm 0.2 / \sqrt{N}$	$0.57 / \sqrt{N}$	$1.23 \pm 0.2 / \sqrt{N}$	$0.37 / \sqrt{N}$

Молепь	2D	EA
TATOTOTIO		

		SRS $(z=0)$		MM (z = 0)	0.7)
Ν	${\gamma}_0$	$\overline{\gamma}$	$\sigma$	$\overline{\gamma}$	σ
5x5x5	$1.376 \pm 0.028$	$1.155 \pm 0.015$	0.059	$1.294 \pm 0.023$	0.039
8x8x8	$1.388 \pm 0.012$	$1.157 \pm 0.009$	0.029	$1.296 \pm 0.010$	0.020
10x10x10	$1.387 \pm 0.005$	$1.154 \pm 0.006$	0.021	$1.292 \pm 0.006$	0.014
20x20x20	нет данных	$1.157 \pm 0.002$	0.007	$1.295 \pm 0.002$	0.005
$N \rightarrow \infty$	$1.38 \pm 0.3 / \sqrt{N}$	$1.15 \pm 0.2 / \sqrt{N}$	$0.65/\sqrt{N}$	$1.29 \pm 0.2 / \sqrt{N}$	$0.44 / \sqrt{N}$

#### Модель SK

		SRS ( <i>z</i> =	=0)	MM (z = 0)	0.7)
Ν	${\gamma}_0$	$\overline{\gamma}$	σ	$\overline{\gamma}$	σ
100	$1.472 \pm 0.038$	$1.325 \pm 0.025$	0.072	$1.413 \pm 0.041$	0.043
200	$1.496 \pm 0.024$	$1.347 \pm 0.014$	0.055	$1.436 \pm 0.026$	0.035
500	$1.510 \pm 0.012$	$1.368 \pm 0.006$	0.036	$1.448 \pm 0.010$	0.023
$N \rightarrow \infty$	$1.51 \pm 0.3 / \sqrt{N}$	$1.36 \pm 0.2 / \sqrt{N}$	$0.76/\sqrt{N}$	$1.44 \pm 0.3 / \sqrt{N}$	$0.47 / \sqrt{N}$
## 4.5 Сравнение с алгоритмами для задачи МАХСИТ

Известной особенностью NP-полных задач является их сводимость друг к другу за полиномиальное по N число шагов. В качестве одного из примеров такого сведения мы приведём здесь взаимосвязь рассмотренной нами задачи квадратичной бинарной оптимизации и задачи о максимальном разрезе графа (MAX-CUT). Суть задачи о максимальном разрезе в следующем: для заданного взвешенного неориентированного графа G(V,W) требуется найти такое разбиение его вершин V на два непересекающихся подмножества  $V_1$  и  $V_2$ , чтобы сумма весов ребёр соединяющих вершины из разных подмножеств была максимальна, т.е.  $\max_{V_1,V_2} \sum_{u \in V_1} \sum_{v \in V_2} w_{uv}$ . Вводя переменные  $s_i = +1$  для вершин находящихся в первом подмножестве и  $s_j = -1$  для вершин во втором

подмножестве, получим

$$\sum_{u \in V_1} \sum_{v \in V_2} w_{uv} = \frac{1}{4} \sum_{i=1}^N \sum_{j=1}^N w_{ij} (1 - s_i s_j),$$

где *N* - число вершин в графе. Как видим, максимизация этого функционала сводится к минимизации квадратичного функционала (1.1), построенного на матрице  $T_{ij} = -w_{ij} / 4$ .

Нам было интересно сравнить эффективность нашего алгоритма MM с алгоритмами, наиболее хорошо зарекомендовавшими себя для решения задачи MAX-CUT [77]. В частности, это алгоритмы: Scatter Search [78] (Marti et al (2009)) и CirCut [79] (Burer et al (2000)). В [77] выложены наборы задач MAX-CUT и результаты для этих алгоритмов, протестированных на данных наборах на компьютере примерно той же вычислительной мощности что и наш (процессор Intel 3.06 GHz).

Сравнение результатов показано в Таблице 4.2. В первом столбце показаны названия матриц. Это матрицы того же сорта, что и матрицы 3D EA с периодическими (тороидальными) граничными условиями. Первые две задачи с гауссовыми связями  $T_{ij}$ , следующие две со связями  $T_{ij} = \pm 1$ .

Размерность задач  $N = L \times L \times L$  при L = 15 и 8. Второй и третий столбцы показывают результаты (значение разреза и время в секундах), приведенные в [77] для алгоритмов Scatter Search и CirCut. В последнем столбце показаны усреднённые значения наилучших разрезов, найденные нашим алгоритмом за  $10^4$  стартов.

	Scatter Search		CirCut		Міх-Маtrix z=0.7, 10 <sup>4</sup> стартов	
Problem	Obj Val	time	Obj Val	time	Obj Val	time
toursg3-15	281 029 888	1023	268 519 648	788	$(2.73 \pm 0.004)*10^8$	8.8
toursg3-8	40 314 704	65	41 684 814	53	$(4.10 \pm 0.010)*10^7$	2.3
tourspm3-15-50	2 964	333	2 895	427	$2\ 889.7\ \pm 4.6$	8.9
tourspm3-8-50	442	48	454	38	$451.9~\pm~1.3$	2.2

Таблица 4.2. Сравнительная таблица по решению 4-х примеров задач МАХСИТ.

Как видно из таблицы, наш алгоритм находит решение, сравнимое с одним из алгоритмов (CirCut или SS) и не более чем на 3% хуже другого, при том что CirCut и SS затрачивают в 20-100 раз больше времени. Наилучший результат оказался для 4-ой задачи, где наш алгоритм также нашёл в некоторых запусках решение 454, соответствующее лучшему из SS и CirCut. Более того, при увеличении числа стартов, удалось найти решение с величиной разреза 456, которое не было до сих пор получено другими алгоритмами.

#### 4.6 Выводы по главе 4

В данной главе был описан алгоритм MM, основанный на микс-матрице (4.1).

Было использование показано. что микс-матрицы приводит К трансформации энергетической поверхности, в результате которой глубокие минимумы становятся существенно глубже, а мелкие – становятся еще мельче или вовсе исчезают. В результате такой трансформации радиус области притяжения любого глубокого минимума V возрастает пропорционально его глубине. Как следствие, экспоненциально по N

возрастает вероятность отыскания этого минимума. На этой основе предложен новый двухэтапный алгоритм случайного поиска MM, который по эффективности и скорости отыскания глобального минимума значительно (экспоненциально по *N*) превышает качество работы стандартного алгоритма случайного поиска.

Следует отметить, что углубление минимума – это вероятностный процесс. Как следует из (4.3) с ростом параметра смешения *z* флуктуации величины  $E_{z0} / E_0$  увеличиваются (см. рис. 4.1). Это означает, что для какойто доли исследуемых матриц трансформация приведет к негативному результату – минимум станет мельче и вероятность его отыскания резко уменьшится. Действительно, такой эффект наблюдался в главе 3, где трансформация поверхности осуществлялась путем возведения исходной матрицы в степень k = 2, 3, 4, 5 (трансформация функционала достигалась при помощи матрицы  $M = T^k$ ). Предложенный в главе 3 алгоритм DDK, несмотря на отличные усредненные характеристики, в случае матриц небольшой размерности (*N* ~ 100) демонстрировал отсутствие стабильности при поиске глобального минимума: для 30% матриц вероятность отыскания глобального минимума снижалась до нуля из-за исчезновения на трансформированном функционале минимума вблизи S<sub>0</sub>. Предложенный здесь алгоритм ММ свободен от таких недостатков по двум причинам. Вопервых, уменьшение параметра z до значения  $z_{opt} \sim 0.5 - 0.7$  приводит к уменьшению флуктуации величины  $E_{z0} / E_0$  и, следовательно, вероятность негативного результата экспоненциально по N уменьшается (напомним, что дисперсия величины  $E_{z0}$  /  $E_0$  спадает как 1 / N). Во-вторых, сдвиг минимума в пространстве при  $z = z_{opt}$  в несколько раз меньше, чем при z = 1, что также положительно сказывается на стабильности и качестве работы алгоритма. Более того, при  $z = z_{ont}$  алгоритм демонстрирует не только высокую стабильность, но и самые высокие свои показатели.

Стандартный алгоритм локальной оптимизации SRS приводит к минимумам, отличающимся по энергии от глобального минимума на 10% для полносвязных матриц (SK) и на 17% для разреженных матриц типа матриц модели Изинга (2D и 3D EA). В то же время алгоритм ММ приводит к нахождению минимумов, отличающихся по энергии от глобального минимума лишь на 4% для полносвязных матриц и на 6.5-7% для разреженных матриц (см. табл. 4.1).

Существенным аргументом в пользу алгоритма ММ является то, что с ростом размерности задачи средняя энергия находимых минимумов не меняется, а происходит лишь сужение дисперсии распределения находимых минимумов (см. рис. 4.5). Это означает, что наш алгоритм даёт гарантированный выигрыш по энергии на 6.6% для полносвязных матриц и на 12% для разреженных матриц по сравнению с SRS. При этом время работы увеличивается менее чем в два раза, поскольку состоит из двух последовательных шагов, на каждом из которых выполняется локальный поиск (не считая затрат на возведение матрицы в квадрат, которое нужно произвести один раз вначале). Отличительной чертой алгоритма MM является его простота.

ММ алгоритм может быть напрямую применён к задаче МАХСUТ (см. раздел 4.5) и показывает высокую конкурентоспособность по сравнению с другими известными алгоритмами для этой задачи. Сравнение с алгоритмами СirCut и Scatter Search показал, что алгоритм ММ даже за 10000 стартов (2с времени для задач с  $N = 8^3$  и 8с для задач с  $N = 15^3$ ) показывает результаты, сравнимые с одним из этих алгоритмов и не более чем на 3% хуже результата другого алгоритма, при том что CirCut и Scatter Search тратят на порядки большее время (см. табл. 4.2).

Ещё раз отметим, что ядром нашего алгоритма является не модификация локального алгоритма поиска, а трансформация энергетической поверхности функционала. Это означает, что алгоритм ММ можно значительно улучшить,

заменив используемую здесь асинхронную динамику модели Хопфилда на более эффективную.

# 4.7 Приложение А.

Для расчета фигурирующей в (4.7) вероятности  $P = \Pr\{s_i^{(0)}h_i^{(z)} < 0 \mid h_i s_i^{(0)} > 0\}$  запишем микс-матрицу в виде:

$$M = (1 - z)(T_0 + T_1) + \frac{z}{\sqrt{N}}(T_0^2 + T_0T_1 + T_1T_0 + T_1^2),$$
(A1)

Тогда величины  $\lambda_i \equiv h_i s_i^{(0)} / \sqrt{N}$  и  $\lambda_i^{(z)} \equiv h_i^{(z)} s_i^{(0)} / \sqrt{N}$  можно представить в виде:

$$\lambda_i = \gamma + \xi_i \tag{A2}$$

$$\lambda_i^{(z)} = (1-z)\lambda_i + z(\gamma\lambda_i + \eta_i)\sqrt{q/(q-1)}$$
(A3)

где

$$\xi_{i} = \frac{1}{\sqrt{N}} \sum_{j \neq i} (T_{1})_{ij} s_{i}^{(0)} s_{j}^{(0)}$$
(A4)

$$\eta_i = \frac{1}{N} \sum_{j \neq i} \left( T_1^2 \right)_{ij} s_i^{(0)} s_j^{(0)}$$
(A5)

- нормально распределенные величины со средними  $\overline{\xi} = 0$  и  $\overline{\eta} = \sigma_{\lambda}^2 - 1$  и стандартными отклонениями  $\sigma_{\xi} = \sigma_{\lambda}$  и  $\sigma_{\eta} = \sigma_{\lambda}^2$ , соответственно. С учётом этого получим:

$$\Pr\left\{\lambda_{i}^{(z)} < 0 \middle| \lambda_{i} > 0\right\} = \frac{1}{\Phi(\gamma / \sigma_{\lambda})} \int_{0}^{+\infty} \frac{dx}{\sqrt{2\pi\sigma_{\lambda}}} e^{-\frac{(x-\gamma)^{2}}{2\sigma_{\lambda}^{2}}} \int_{-\infty}^{-\alpha x} \frac{dy}{\sqrt{2\pi\sigma_{\eta}}} e^{-\frac{(y-\eta)^{2}}{2\sigma_{\eta}^{2}}}$$
(A6)

где  $\alpha = \left[ (1-z)\sqrt{(q-1)/q} + z\gamma \right] / z$ . Отсюда заменой переменных интегрирования получаем выражение (4.7).

# 5. ОГРАНИЧЕНИЯ НА ВЕСА ПАТТЕРНОВ В КВАЗИХЕББОВСКОЙ МАТРИЦЕ (ДОПОЛНИТЕЛЬНАЯ ГЛАВА)

В главе 2 мы показали, что любая матрица может быть представлена в виде квазихеббовского разложения по конфигурациям минимумов. При этом конфигурация может быть минимумом, только если её вес в разложении не меньше некоторого критического значения  $r_c$ . К сожалению, теоретически оценить значение  $r_c$  не удаётся. С другой стороны, можно пойти обратным путём и рассмотреть квазихеббовскую матрицу, *построенную* на случайных паттернах. В этом случае, пользуясь методами статистической физики, удаётся получить различные асимптотические формулы, в том числе для ёмкости памяти и связанной с ней критическим весом  $r_c$ , что и делается в данной главе.

#### 5.1 Квазихеббовская модель

Здесь мы рассмотрим проблему устойчивости спектра минимумов квадратичного функционала:

$$E(\mathbf{S}) = -\sum_{i=1}^{N} \sum_{j=1}^{N} W_{ij} s_i s_j \quad , \tag{5.1}$$

Состояние системы, описываемой функционалом  $E(\mathbf{S})$ , задается *N*мерным вектором  $\mathbf{S} = (s_1, s_2, ..., s_N)$  с бинарными координатами  $s_i = \pm 1$ . Такие векторы будем называть конфигурационными (конфигурациями), а функционал  $E = E(\mathbf{S})$  - энергией состояния. Без потери общности матрицу *W* будем считать симметричной.

В общем случае почти ничего неизвестно ни о числе, ни об устройстве локальных минимумов функционала (5.1). Относительно ясное представление об энергетической поверхности получено лишь для специфических матриц модели Шеррингтона-Киркпатрика спинового стекла [37] (Sherrington and Krkpatrick (1975)) и высоко симметричной нейронной сети Хопфилдова типа [80] (Литинский (1999).

полный анализ энергетической поверхности проведен Более В [73], [81], [82] (Amit et al (1987), Amit et al (1985), Hertz et al (1991)) методами статистической физики для функционала E = E(S), базирующегося на матрице Хебба - корреляционной матрице вида (5.2) (см. ниже), построенной в виде суммы внешних произведений М случайных конфигурационных  $\xi^{\mu} = (\xi^{\mu}_{1}, \xi^{\mu}_{2}, ..., \xi^{\mu}_{N}), \qquad \xi^{\mu}_{i} = \pm 1, \qquad \mu = 1, 2, ..., M$ , называемых векторов *паттернами*. Проведенный в [73],[81] анализ показал: если паттерны  $\xi^{\mu}$ случайны и независимы, и их число не слишком велико (M < 0.138N), то в небольшой окрестности каждого паттерна  $\xi^{\mu}$  непременно имеется локальный минимум функционала (5.1). При превышении критического значения  $M \approx 0.138 N$  в системе происходит фазовый переход первого рода, приводящий к полной перестройке спектра минимумов: минимумы скачком отдаляются от паттернов и перекрытие паттерна с ближайшим минимумом стремится к нулю.

До недавнего времени считалось, что методы статистической физики можно использовать только для Хеббовских матриц связи. Однако как было показано в работе [72] и отмечалось в главе 2, любую симметричную матрицу W можно представить в виде *квази-хеббовского* разложения по некоррелированным конфигурациям  $\xi^{\mu}$ :

$$W_{ij} = \sum_{\mu=1}^{M} r_{\mu} \,\xi_{i}^{\mu} \,\xi_{j}^{\mu} \tag{5.2}$$

где число *M* определяется точностью аппроксимации исходной матрицы *W*. Квази-хеббовским представление (5.2) называется, поскольку весовые множители  $r_{\mu}$  различны; при  $r_{\mu} = 1$  матрица (5.2) является Хэббовской.

Представление (5.2) позволяет использовать методы статистической физики для анализа произвольных матриц связи.

#### 5.2 Основное уравнение

С учетом (5.2) энергию в состоянии S представим в виде:

$$E(\mathbf{S}) = -\sum_{\mu=1}^{M} r_{\mu} m_{\mu}^{2} \quad , \quad m_{\mu}(\mathbf{S}) = \frac{1}{N} \sum_{i=1}^{M} s_{i} \xi_{i}^{\mu}$$
(5.3)

где  $m_{\mu} = m_{\mu}(\mathbf{S})$  - перекрытие состояния **S** с паттерном  $\xi^{\mu}$ . Методы статистической физики позволяют получить уравнения для перекрытий локального минимума функционала (5.1) с паттернами  $\xi^{\mu}$ . Решив эти уравнения можно понять, при каких условиях перекрытие локального минимума с *k*-м паттерном будет порядка 1 ( $m_k \sim 1$ ). Иными словами: при каких условиях локальный минимум совпадает (или почти совпадает) с паттерном  $\xi^k$ .

Анализ проведем в асимптотическом пределе  $N \to \infty$ , полагая пропорционально большим и число паттернов  $M = \alpha N$ . Коэффициент пропорциональности  $\alpha$  в теории нейронных сетей называют *параметром загрузки*. Пусть состояние **S** соответствует минимуму энергии и этот локальный минимум близок к паттерну  $\xi^k$ . Тогда, повторяя вычисления, проделанные в [73],[81],[82], получим систему уравнений:

$$m_{k} = \operatorname{erf}\left(\frac{r_{k}m_{k}}{\sqrt{2}\sigma_{k}}\right), \quad \sigma_{k}^{2} = \frac{1}{N}\sum_{\mu\neq k}^{M}\frac{r_{\mu}^{2}}{\left(1-C\,r_{\mu}\right)^{2}}, \quad C = \frac{\sqrt{2}}{\sigma_{k}\sqrt{\pi}}\exp\left[-\left(\frac{r_{k}m_{k}}{\sqrt{2}\sigma_{k}}\right)^{2}\right]$$
(5.4)

Вводя новую переменную  $y = r_k m_k / \sqrt{2}\sigma_k$  и исключая из (5.4) величины  $\sigma_k$  и *C*, получим *основное уравнение* для *k*-го паттерна:

$$F(y) = \alpha , \quad F(y) = \gamma^{2} \left[ \frac{1}{M} \sum_{\mu \neq k}^{M} \frac{r_{\mu}^{2}}{\left(r_{k} \cdot \phi - r_{\mu}\right)^{2}} \right]^{-1}$$
(5.5)

где  $\gamma = \gamma(y)$  и  $\phi = \phi(y)$  - монотонно убывающая и монотонно возрастающая функции:

$$\gamma = \sqrt{\frac{2}{\pi}} e^{-y^2}, \quad \phi = \frac{\sqrt{\pi}}{2} \cdot \frac{\text{erf } y}{y} e^{y^2} .$$
 (5.6)

Решение (5.5) дает основные характеристики энергетической поверхности: энергию локального минимума (5.3), его перекрытие  $m_k = \operatorname{erf} y$  с ближайшим паттерном, критическое значение ( $\alpha_c$ ) параметра  $\alpha$ : решение уравнения (5.5) существует только при  $\alpha \leq \alpha_c$ . Отметим, что уравнение (5.5) возникает в результате минимизации *свободной энергии* (см. [82]). Из всех возможных решений (5.5) минимуму свободной энергии отвечает решение с максимальным значением перекрытия  $m_k$ , остальные решения - ложные. Поэтому, необходимо выбирать то решение уравнения  $\alpha = F(y)$ , которое лежит максимально справа по оси абсцисс. Этим принципом мы будем широко пользоваться в дальнейшем.



Рис. 5.1. Графическое решение уравнения (5.7). Кривая 1 - вид F(y) при  $r \le 1$ : нижняя пунктирная прямая отвечает загрузке  $\alpha = 0.03$ ,  $y_{\alpha}$  - истинное решение,  $y'_{\alpha}$  - ложное решение,  $y_{max}$  - точка максимума F(y). Кривая 2 - типичный вид F(y) при  $r > r_m$ :  $y_0$  - нуль F(y),  $\alpha_c(r)$  - критическое значение загрузки. Масштаб кривой 2 слегка уменьшен, чтобы она не сливалась с кривой 1.

Сказанное продемонстрируем на примере случая одинаковых весов  $(r_{\mu} \equiv 1, \forall \mu = \overline{1, M})$ . В этом случае система (5.4) сводится к хорошо известным уравнениям для модели Хопфилда [82]. Поскольку все паттерны здесь равноправны, индекс *k* в уравнениях (5.4)-(5.5) можно опустить и основное уравнение (5.5) примет вид

$$F(y) = \alpha$$
,  $F(y) = \gamma^2 (\phi - 1)^2$ . (5.7)

Эта функция имеет колоколообразную форму (рис. 5.1) с единственным максимумом  $F_{max} \approx 0.138$ , достигаемым при  $y = y_{max} \approx 1.511$ . Соответственно, уравнение  $F(y) = \alpha$  не имеет решения при загрузке  $\alpha$ , большей критического значения  $\alpha_c = F_{max}$ . При  $\alpha < \alpha_c$  имеется два решения в точках  $y'_{\alpha} < y_{max}$  и  $y_{\alpha} > y_{max}$ . Интерес представляет наибольшее решение  $y_{\alpha}$ , соответствующее минимуму свободной энергии; решение у' является ложным и его следует отбросить. С ростом  $\alpha$  решение  $y_{\alpha}$  смещается к  $y_{max}$ и перекрытие  $m = \operatorname{erf} y_{\alpha}$  постепенно уменьшается (локальный минимум отдаляется от паттерна). Когда загрузка достигает критической величины  $\alpha = \alpha_c$ , ложное и истинное решения сливаются в точке  $y = y_{max}$  и перекрытие принимает критическое значение  $m_c \approx 0.967$ . При превышении критической загрузки ( $\alpha > \alpha_c$ ) происходит срыв решения: перекрытие скачком падает до нуля. В терминах статистической физики это означает, что в системе произошел фазовый переход первого рода. В терминах нейронных сетей это означает развал ассоциативной памяти. Фазовый переход сопровождается полной перестройкой спектра локальных минимумов. Эта хорошо известная неустойчивость системы впервые рассмотрена в [73],[81].

#### 5.3 Простейший случай.

Другого типа неустойчивость системы рассмотрим ниже на примере, когда все веса  $r_{\mu}$  кроме одного равны 1, и только один весовой множитель отличается от остальных:

$$r_1 = r$$
,  $r_2 = r_3 = \dots = r_M = 1$ . (5.8)

Казалось бы, отличия от канонической модели Хопфилда невелики: огромное число паттернов участвует в образовании матрицы W с одним и тем же (вырожденным) весом, и только один паттерн дает другой вклад. Интуиция подсказывает, что влияние единственного паттерна с индивидуальным весом  $r \neq 1$  должно быть пренебрежимо мало по сравнению с влиянием бесконечно большого числа паттернов. Оказывается, однако, что интуиция нас обманывает: паттерн с индивидуальным весом r может существенно изменить спектр локальных минимумов. Рассмотрим по отдельности ситуацию для паттерна с индивидуальным весом  $r \neq 1$  и для паттернов, с многократно вырожденным весовым множителем 1.

5.3а) Паттерны с вырожденным весом. Рассмотрим случай для одного из паттернов  $\xi^k$  с вырожденным весом  $r_k = 1$ ,  $k = \overline{2, M}$ ; поскольку все они равноправны индекс *k* будем опускать. Для этого случая из (5.5) получим:

$$F(y) = \frac{\gamma^2 (\phi - 1)^2 (\phi - r)^2}{(\phi - r)^2 + \varepsilon r^2 (\phi - 1)^2}, \quad \text{где} \quad \varepsilon = M^{-1} \to 0.$$
 (5.9)

Анализ уравнения  $F(y) = \alpha$  проведем при малом, но конечном значении параметра  $\varepsilon$ , устремляя его затем к нулю. При  $r \le 1$  выражение (5.9) в пределе  $\varepsilon \to 0$  совпадает с (5.7). При r > 1 в кривой F(y) появляется асимптотически узкий (ширины  $\sim \varepsilon^{1/3} \to 0$ ) провал до нуля в точке, где  $\phi(y) = r$ . С ростом r этот провал сдвигается вправо по оси y и при  $r = r_m$  $(r_m \approx 5.568)$  достигает точки  $y = y_{max}$ . При  $1 < r < r_m$  провал находится области  $0 < y < y_{max}$ , где возможны только ложные решения. При этом, в области  $y > y_{max}$ , доставляющей истинное решение уравнения  $F(y) = \alpha$ , никаких изменений в форме F(y) не происходит. Сказанное означает, что при  $r \le r_m$ нет никаких различий с рассмотренным выше каноническим случаем равных весов:  $\alpha_c \approx 0.138$  и  $m_c \approx 0.967$ .

Влияние веса *r* начинает сказываться на свойствах системы, когда его величина превышает критическое значение  $r > r_m$ . В этом случае нуль функции F(y) - точка  $y_0$ , определяемая уравнением  $\phi(y_0) = r$  - находится правее максимума  $y_{max}$  (см. рис. 5.1). Истинное решение уравнения  $F(y) = \alpha$ может иметь место только при  $y > y_0$ . Соответственно, критическое значение загрузки определяется максимумом F(y) в области  $y > y_0$ . Этот максимум вплотную (на расстоянии  $\varepsilon^{1/3} \rightarrow 0$ ) примыкает к точке  $y_0$  справа, а значение функции в максимуме имеет вид  $F_{\text{max}}^{(0)} = (r-1)^2 \cdot \gamma(y_0)^2$ . Для  $r \ge r_m$  в приближении r >> 1 можно получить соотношение  $y_0 \approx \sqrt{\ln 2r}$ , а для критических значений загрузки  $\alpha_c(r) = F_{\text{max}}^{(0)}$  и перекрытия  $m_c(r) = \text{erf } y_0$  будут справедливы оценочные выражения:

$$\alpha_{c}(r) = \frac{(r-1)^{2}}{2r^{2}\ln 2r} , \quad m_{c}(r) = 1 - \frac{1}{2r\ln 2r}$$
(5.10)

При  $r = r_m$ , как и следовало ожидать, получим  $\alpha_c(r) \approx 0.138$  и  $m_c(r) \approx 0.967$ . С ростом веса *r* точка  $y_0$  медленно сдвигается вправо по оси абсцисс. При этом критическая загрузка спадает до нуля, а срыв решения наступает при достаточно большом перекрытии  $m_c(r) \rightarrow 1$ .

5.36) Паттерн с индивидуальным весом. Рассмотрим паттерн  $\xi^1$  с индивидуальным весом  $r \neq 1$  и его перекрытие  $\overline{m}$  с ближайшим к нему локальным минимумом (все характеристики, связанные с данным паттерном, будем обозначать чертой сверху). Функция F(y) в этом случае принимает вид:

$$F(y) = \gamma^{2} (\phi \cdot r - 1)^{2} , \qquad (5.11)$$

Анализ (5.11) показывает, что при  $r \rightarrow 0$  критическая загрузка (для данного паттерна) также стремится к нулю. При r = 1 выражение (5.11) совпадает с (5.7) и результаты анализа совпадают с классическими.

Несколько неожиданный результат получается при r > 1. В этом случае точка "срыва"  $\overline{y}_m$ , в которой F(y) достигает максимума, определяется из уравнения

$$\phi(y) = 1 + \frac{2y^2}{r} . \tag{5.12}$$

Отыскав  $\bar{y}_m$  можно вычислить критические характеристики  $\bar{m}_c = \text{erf } \bar{y}_m$ и  $\bar{\alpha}_c = F(\bar{y}_m)$ . Легко проверяется, что уравнение (5.12) имеет нетривиальное решение только при r < 3, а в случае  $r \ge 3$  имеется только тривиальное решение  $\bar{y}_m = 0$ . Базовое уравнение  $F(y) = \alpha$  в данном случае имеет два типа решений. При r < 3 имеет место обычный срыв решения: при превышении критического значения загрузки  $\alpha = \overline{\alpha}_c$  перекрытие скачком падает от значения  $\overline{m}_c$  до нуля. При  $r \ge 3$  нет срыва решения: с ростом  $\alpha$  перекрытие  $\overline{m}$  плавно стремится к нулю. Критическая загрузка  $\overline{\alpha}_c = \overline{\alpha}_c(r)$ , при достижении которой перекрытие  $\overline{m}$  обращается в ноль, в случае  $r \ge 3$ задается выражением:

$$\bar{\alpha}_{c} = \frac{2}{\pi} (r-1)^{2}$$
(5.13)

Как видим, для значений  $r \ge 3$  фазовый переход первого рода, до того существовавший в системе, пропадает. Тот факт, что критическая загрузка может обращаться в нуль не должен вводить в заблуждение: если зафиксировать величину загрузки, то с ростом веса r перекрытие индивидуального паттерна с его локальным минимумом будет асимптотически стремиться к 1.

5.36) Обсуждение. В отличие от классического подхода [73],[81], проведенный выше анализ позволяет проследить за тем, как поведет себя система, если зафиксировать загрузку  $\alpha$  и варьировать вес r.

Рассмотрим сначала поведение индивидуального паттерна  $\xi^1$ . Анализ (5.5) показывает, что при заданной загрузке  $\alpha$  существует некоторое критическое значение веса  $\overline{r}_c = \overline{r}_c(\alpha)$ : при  $r < \overline{r}_c$  имеем  $\overline{m} = 0$ , перекрытие отлично от нуля только при  $r \ge \overline{r}_c$ . Характер зависимости  $\overline{m} = \overline{m}(r)$  существенным образом зависит от величины параметра  $\alpha$ . При  $\alpha < 8/\pi$  будет  $\overline{r}_c < 3$  и имеет место следующее поведение: при  $r < \overline{r}_c$  имеем  $\overline{m} = 0$ ; при достижении критического значения  $r = \overline{r}_c$  перекрытие скачком возрастает от нуля до некоторого значения  $\overline{m}_c = \overline{m}(r_c)$ ; дальнейший рост r сопровождается плавным ростом перекрытия  $\overline{m} \rightarrow 1$ . При  $\alpha \ge 8/\pi$  критический вес задается выражением  $\overline{r}_c = 1 + \sqrt{\pi\alpha/2}$  и в поведении перекрытия нет скачка:  $\overline{m} = 0$  при  $r < \overline{r}_c$ ; при превышении критического значения  $\overline{r}_c$  лерекрытие с ростом r плавно нарастает от нуля до 1 (при r >>1).

Полученные результаты проверялись в компьютерном эксперименте. Для размерности задачи N = 30000 фиксировали загрузку  $\alpha$ , генерировали  $M = \alpha N$  случайных паттернов и для них строили матрицу связи (5.2). Для каждого значения  $\alpha$  перекрытие  $\overline{m}$  усреднялось по ансамблю различных матриц. Результаты эксперимента приведены на рис. 5.2. Для испытаний были выбраны три значения  $\alpha$ . Для двух из них  $\alpha = 0.12$  ( $\overline{r_c} = 0.944$ ,  $\overline{m_c} = 0.971$ ) и  $\alpha = 0.38$  ( $\overline{r_c} = 1.501$ ,  $\overline{m_c} = 0.919$ ) на экспериментальных кривых явственно наблюдается ожидаемый резкий скачок перекрытия  $\overline{m}$  в районе расчетных критических значений. Тот факт, что скачок происходит не от нулевого значения, связан с конечной размерностью задачи. Кроме того, теория не корректна при малом перекрытии ( $\overline{m} \rightarrow 0$ ).



Рис. 5.2. Наблюдаемая в эксперименте зависимость  $\overline{m} = \overline{m}(r)$  :  $\overline{m}$  - перекрытие индивидуального паттерна с ближайшим локальным минимумом, r - весовой множитель индивидуального паттерна. Графики построены для размерности N = 30000 при трёх значениях параметра загрузки  $\alpha = 0.12$ , 0.38 и 3.0 соответственно. Сплошные линии - эксперимент, штриховые линии – теория.

Для третьего значения параметра загрузки  $\alpha = 3$  ( $\bar{r}_c = 3.171$ ,  $\bar{m}_c = 0.919$ ) ожидалось плавное возрастание перекрытия от 0 до 1. Действительно, соответствующая кривая на рис. 5.2 возрастает плавно, без скачков. Однако, согласно теории, это возрастание должно начинаться с  $\bar{r}_c = 3.171$ , в то время как экспериментальная кривая отличается от нуля уже гораздо раньше. Это расхождение мы объясняем высказанными выше соображениями.

Обратимся теперь к случаю вырожденного паттерна. Будем полагать, что значение загрузки зафиксировано на уровне  $\alpha \le \alpha_c \approx 0.138$  (при  $\alpha > \alpha_c$ уравнение (5.5) имеет только тривиальное решение m=0). Рассмотрим, как с ростом r изменяется перекрытие m=m(r). Анализ показывает, что существует критическое значение величины r, задаваемое выражением  $r_c = \phi(y_{\alpha})$ , где  $y_{\alpha}$  - решение уравнения (5.7). При изменении величины r от 0 до  $r_c$  перекрытие остается неизменным с классическим значением  $m = \text{erf } y_{\alpha}$ . При превышении значения  $r = r_c$  перекрытие скачком обращается в нуль.

Для проверки существования такого типа неустойчивости проведен ряд экспериментов. На рис. 5.3 приведены результаты эксперимента, в ходе которого измерялась зависимость m = m(r) при фиксированной загрузке  $\alpha = 0.12$  и нескольких значениях размерности N (N = 3000, 10000 и 30000). Согласно теории, для  $\alpha = 0.12$  "срыв" перекрытия *m* в асимптотическом пределе  $M = \alpha N \rightarrow \infty$  должен происходить при r = 17.1. На рис. 5.3 это место отмечено правой вертикальной пунктирной прямой с пометкой " $M = \alpha N \rightarrow \infty$ ". Зависимость m = m(r), которая реально наблюдалась в эксперименте, показана на рис. 5.3 тремя сплошными кривыми. Кажущееся заметное отличие теории от эксперимента не должно нас смущать, поскольку оно обусловлено конечной размерностью матрицы. Приведенные выше теоретические результаты получены в термодинамическом пределе  $M = \alpha N \rightarrow \infty$ . Однако, можно учесть поправку на конечность размерности, введя в формулу (5.12) реальное значение множителя  $\varepsilon = M^{-1}$ . Например, при N = 30000 и  $\alpha = 0.12$  имеем  $M = \alpha N = 3600$ . Подставив соответствующий множитель в уравнение (5.12), получаем, что "срыв" перекрытия должен происходить не в районе r = 17.1, а много раньше - при r = 7.1. Соответствующая пунктирная прямая с пометкой "М = 3600" показана на рис. 5.3; ее положение значительно лучше коррелирует с поведением экспериментальных кривых.



**Рис. 5.3**. Зависимость m = m(r): m - перекрытие паттерна с вырожденным весовым множителем с ближайшим локальным минимумом, r - весовой множитель индивидуального паттерна. Сплошные линии – результаты экспериментов, проведенных при фиксированной загрузке  $\alpha = 0.12$  для трех значений N. Штриховые вертикальные прямые – ожидаемые места "срыва" перекрытия, вычисленные при конечной размерности N = 30000 (левая вертикаль) и в асимптотическом пределе  $N \rightarrow \infty$  (правая вертикаль).

5.3г) Глубина локальных минимумов. Выясним, как глубина локальных минимумов зависит от величины весового множителя r. Под *глубиной* минимума будем понимаем модуль значения энергетического функционала |E|. Сравнивать глубины минимумов для различных матриц можно только когда энергии вычисляются в одной и той же шкале. Мы нормировали элементы каждой матрицы на корень квадратный из дисперсии матричных элементов:  $\sigma_w = \sqrt{r^2 + \alpha N}$  - стандартное отклонение матричных элементов квази-Хеббовской матрицы вида (5.2). (Среднее значение матричных элементов равно 0.) Для глубины локального минимума получаем выражение:

$$|E|=\frac{1}{\sigma_W}\sum_{\mu=1}^M r_\mu m_\mu^2.$$

Используя это выражение, получаем для локального минимума, находящегося в окрестности паттерна  $\xi^1$  с индивидуальным весом  $r_1 = r$ :

$$|E_1| = \frac{1}{\sigma_W} (r m_1^2 + \sigma_1^2 - \alpha)$$
,  $\sigma_1 = r \gamma \varphi$ .



**Рис. 5.4а.** Зависимость от *r* глубины локального минимума из окрестности паттерна  $\xi^1$  с индивидуальным весом  $r_1 = r$ . Загрузка  $\alpha$  меняется от 0.001 до 2.



Рис. 5.46. Зависимость от *r* глубины локального минимума из окрестности одного из паттернов  $\xi^{\mu}$  с одинаковыми весами  $r_{\mu} = 1$  ( $\mu \ge 2$ ). Загрузка  $\alpha$  меняется от 0.001 до 0.137.

Здесь  $\alpha$  - значение параметра загрузки,  $\gamma = \gamma(y)$  и  $\varphi = \varphi(y)$  даются выражениями (5.6). Анализ и численные расчеты показывают, что с ростом rглубина минимума монотонно растет. Пока r меньше некоторого критического значения  $r_c(\alpha)$ , локальный минимум далек от паттерна  $(m_1 \approx 0)$ , а его глубина равна нулю. При значениях  $r > r_c(\alpha)$  минимум резко приближается к паттерну  $(m_1 \rightarrow 1)$ , а его глубина почти скачком увеличивается. После чего  $|E_1|$  растет как  $r / \sqrt{r^2 + \alpha N}$  - см. рис. 5.4a, где для различных значений параметра загрузки  $\alpha$  показана зависимость  $|E_1|$  от r. Когда r становится очень большим  $(r > \sqrt{\alpha N})$  глубина минимума асимптотически стремится к 1. Действительно, при столь большом r можно считать, что матрица W построена на одном-единственном паттерне  $\xi^1$ .

Рассмотрим теперь локальные минимумы, связанные с паттернами, имеющими один и тот же весовой множитель  $r_{\mu} = 1$  ( $\mu \ge 2$ ). Во-первых, независимо от r, такие локальные минимумы существуют только когда  $\alpha \le \alpha_c$ . Выражение для глубины  $\mu$ -го минимума имеет вид:

$$\left|E_{\mu}\right| = \frac{1}{\sigma_{W}}\left(m_{\mu}^{2} + \sigma_{\mu}^{2} - \alpha\right),$$
 где  $\sigma_{\mu} = \gamma \varphi$ .

Анализ этого выражения показывает, что с ростом *r* глубина локального минимума уменьшается - см. рис. 5.4б. Когда *r* станет равным некоторому критическому значению  $r_c(\alpha)$ , глубина минимума достигнет наименьшего значения  $E_c = E_c(\alpha)$ :

$$E_c = \frac{2}{\sqrt{\alpha_c N}} - \alpha_c \alpha$$

При дальнейшем увеличении r перекрытие  $m_{\mu}$  скачком обращается в ноль: минимум резко «уходит» от паттерна  $\xi^{\mu}$ , а его глубина  $|E_{\mu}|$  скачком падает до нуля. С точки зрения ассоциативной памяти это означает, что при  $r > r_c(\alpha)$  память сети разрушается.

### 5.4 Произвольное распределение весов r<sub>и</sub>

Пусть теперь все веса  $r_{\mu}$  в уравнении (5.5) различны. Без ограничения общности можно упорядочить их по убыванию:

$$r_1 > r_2 > r_3 > \dots \ge 0$$
.

В стандартной модели Хопфилда, когда все веса равны, паттерны полностью равноправны. Условия, при которых в окрестности паттерна существует локальный минимум, одинаковы для всех паттернов. Это позволяет при формулировке конечных результатов использовать такую характеристику, как число паттернов M. Напротив, когда веса  $r_{\mu}$  различны, индивидуализированы и условия, при которых в окрестности того или иного паттерна существует локальный минимум. Теперь все будет определяться не числом паттернов, а распределением весов  $r_{\mu}$ . Можно избавиться от лишнего параметра M, сократив на него обе части уравнения (5.5) и суммируя правую часть уравнения (5.5) до бесконечности. Фактически, мы переходим к рассмотрению модели с бесконечным числом паттернов:  $M = \infty$ . Уравнение (5.5) принимает вид:

$$N = \sum_{\mu=1 \neq k}^{\infty} f_{\mu}^{(k)}(y), \qquad (5.14)$$

где

$$f_{\mu}^{(k)}(y) = \left(\frac{t_{\mu}^{(k)}}{\gamma(\varphi - t_{\mu}^{(k)})}\right)^2, \quad \mathbf{M} \quad t_{\mu}^{(k)} = \frac{r_{\mu}}{r_k}, \quad \mu = 1, 2, \dots, k - 1, k + 1, k + 2, \dots.$$
(5.15)

Уравнение (5.14) связывает между собой размерность N и номер паттерна k, а такая характеристика, как параметр загрузки  $\alpha = M / N$ , отсутствует.

Величины  $t_{\mu}^{(k)}$  упорядочены по убыванию. Обратим внимание на то, что первые k-1 из них превосходят 1, а остальные меньше 1:

$$t_1^{(k)} > t_2^{(k)} > \dots > t_{k-1}^{(k)} > 1 > t_{k+1}^{(k)} > t_{k+2}^{(k)} > \dots$$

Правая часть уравнения (5.14) есть результат суммирования по набору функций  $\{f_{\mu}^{(k)}(y)\}_{\mu\neq k}$ . Поведение функции  $f_{\mu}^{(k)}(y)$  существенно зависит от того, больше или меньше 1 константа  $t_{\mu}^{(k)}$  в знаменателе. Если  $t_{\mu}^{(k)} < 1$ , функция  $f_{\mu}^{(k)}(y)$  непрерывна и ограничена. Если же  $t_{\mu}^{(k)} > 1$ , функция  $f_{\mu}^{(k)}(y)$ имеет точку сингулярности. Поясним это подробнее.

При  $y \to \infty$ , независимо от величины  $t_{\mu}^{(k)}$  знаменатель  $\gamma(\varphi - t_{\mu}^{(k)})$ стремится к 0, оставаясь все время положительным - это следует из анализа, проделанного в предыдущих разделах. Таким образом, каждая функция  $f_{\mu}^{(k)}(y)$  на бесконечности неограниченно возрастает. Следовательно, правая часть уравнения (5.14) на бесконечности неограниченно растет. Проанализируем теперь поведение функции  $f_{\mu}^{(k)}(y)$  для конечных значений аргумента. Поскольку функция  $\varphi(y)$  в знаменателе больше 1, для  $t_{\mu}^{(k)} < 1$ знаменатель функции  $f_{\mu}^{(k)}(y)$  нигде в ноль не обращается. В этом случае  $f_{\mu}^{(k)}(y)$  непрерывна и ограничена при любом значении аргумента. Иначе обстоит дело, когда  $t_{\mu}^{(k)} \ge 1$ . В этом случае знаменатель функции  $f_{\mu}^{(k)}(y)$  обращается в ноль при некотором значении аргумента  $y_{\mu}^{(k)}$ :

$$\varphi(y_{\mu}^{(k)}) = t_{\mu}^{(k)} \Leftrightarrow y_{\mu}^{(k)} = \varphi^{-1}(t_{\mu}^{(k)});$$

здесь  $\varphi^{-1}$  означает отображение, обратное к  $\varphi$ . В точке  $y_{\mu}^{(k)}$  функция  $f_{\mu}^{(k)}(y)$  терпит разрыв второго рода. Напомним, что первые k - 1 величин  $t_{\mu}^{(k)}$  больше 1. Легко сообразить, что положительная функция в правой части уравнения (5.14) имеет разрывы второго рода в точках  $y_{k-1}^{(k)} < y_{k-2}^{(k)} < ... < y_1^{(k)}$ , а на бесконечности неограниченно возрастает.

Для удобства преобразуем уравнение (5.14), перейдя к обратным величинам:

$$\frac{1}{N} = F_k(y),$$
 где  $F_k(y) = \frac{1}{\sum_{\mu \neq k}^{\infty} f_{\mu}^{(k)}(y)}.$  (5.16)

С учетом сказанного очевидно, что неотрицательная функция  $F_k(y)$  в правой части уравнения (5.16) обращается в ноль в точках  $y_{k-1}^{(k)} < y_{k-2}^{(k)} < ... < y_1^{(k)}$ , а на бесконечности стремится к нулю. Правее самого правого нуля  $y_1^{(k)}$ , там, где выполняется неравенство  $\varphi(y) > t_1^{(k)}$ , функция  $F_k(y)$  сначала возрастает, а затем, пройдя через максимум, монотонно стремится к нулю (см. рис. 5.5). Обозначим самую правую точку максимума  $F_k(y)$  через  $y_c^{(k)}$ :  $y_1^{(k)} < y_c^{(k)}$ . Этим значением аргумента определяются критические характеристики, связанные с распознаванием k-го паттерна. Точка  $y_c^{(k)}$  играет ту же роль, что и критическая точка  $y_c$  в стандартной модели Хопфилда.

Дело в том, что уравнение (5.16) имеет несколько решений – столько, сколько раз функция  $F_k(y)$  пересекается с прямой линией, параллельной оси абсцисс на высоте 1/N. Нас интересует только то пересечение, которое лежит

правее точки максимума  $y_c^{(k)}$  - именно этому решению уравнения (5.16) отвечает минимум свободной энергии.

Сказанное справедливо для произвольных весов  $\{r_{\mu}\}_{\mu=1}^{\infty}$ . Для конкретного набора весов этот анализ можно детализировать. Мы рассмотрим два таких конкретных распределения.

5.4а) Веса в виде гармонической последовательности:  $r_{\mu} = 1 / \mu$ .

В этом случае,

$$f_{\mu}^{(k)}(y) \sim \left(\frac{k}{\mu \varphi - k}\right)^2 = \frac{a_k^2}{(\mu - a_k)^2},$$
 где  $a_k \equiv a_k(y) = \frac{k}{\varphi(y)} < 1.$ 

Следовательно, можно записать:

$$\sum_{\mu \neq k}^{\infty} f_{\mu}^{(k)}(y) = \frac{a_k^2}{\gamma^2(y)} \sum_{\mu \neq k}^{\infty} \frac{1}{(\mu - a_k)^2}.$$
(5.17)

Сумма в правой части выражения (23) напоминает дзета-функцию Гурвица [83] (Бейтмен, Эрдейи (1965))  $\zeta(s \, a \, для \, з$ начения первого аргумента, равного 2:

$$\zeta(s,a) = \sum_{\mu=0}^{\infty} \frac{1}{(\mu+a)^s}$$

По традиции, поведение дзета-функции изучают при положительных значениях второго аргумента *a*, в то время как у нас в (5.17) этот аргумент по построению отрицателен. Нетрудно показать, что для  $a \in (0,1)$  справедливо:  $\zeta(2,-a) = \frac{1}{a^2} + \zeta(2,1-a)$ . После несложных преобразований придадим уравнению (5.16) окончательный вид:

$$\frac{1}{N} = \frac{\gamma^2 (\varphi - 1)^2}{a_k^2 (\varphi - 1)^2 \zeta (2, 1 - a_k) - 1}.$$
(5.18)

Здесь  $\varphi(y)$  и  $\gamma(y)$  даются выражениями (6),  $a_k \equiv a_k(y) = k/\varphi(y)$  и поведение правой части уравнения (5.18) – будем по-прежнему обозначать

эту функцию через  $F_k(y)$  - интересует нас для тех значений аргумента, которые превосходят самый правый максимум  $F_k(y)$ :  $a_k(y) < 1 \Leftrightarrow \varphi(y) > k$ .

На рис. 5.5 показано, как ведет себя правая часть уравнения (5.18) для паттерна с номером 5: k = 5. Функция  $F_5(y)$  обращается в ноль в точках  $y_4^{(5)} < y_2^{(5)} < y_2^{(5)} < y_1^{(5)}$ . Для значений аргумента y, превосходящих наибольшую из них  $y_1^{(5)}$ , функция  $F_5(y)$  сначала возрастает до максимума в точке  $y_c^{(5)}$ , а затем монотонно стремится к нулю. Пунктирная прямая линия, параллельная оси абсцисс, проведена на высоте 0.001. Приравняв к этому значению левую часть уравнения (5.18), получим N = 1000. Иными словами, для матрицы такой размерности в окрестности 5-го паттерна непременно существует локальный минимум функционала, и его перекрытие с паттерном близко к 1:  $y_0 \approx 3.5$ , а перекрытие  $m_5 = \operatorname{erf}(y_0)$ .



**Рис. 5.5**. Поведение правой части уравнения (24) для k = 5 (сплошная линия):  $y_0$  - решение уравнения (5.18) для 1/N = 0.001,  $y_c^{(5)}$  - критическое значение аргумента *у* (см. текст).

Станем постепенно уменьшать размерность N. Прямая линия поднимется вверх, а решение  $y_0$  уравнения (5.18) сместится влево, в область меньших значений. Так будет продолжаться до тех пор, пока  $y_0$  не достигнет критического значения  $y_c^{(5)}$ . Этим и определится минимальная размерность  $N_{\min}$ , при которой в окрестности 5-го паттерна существует локальный минимум. Для меньших размерностей,  $N < N_{\min}$ , локального минимума в

окрестности 5-го паттерна не существует. На языке нейронных сетей это означает, что при  $N < N_{min}$  5-й паттерн сетью не распознается.

В приведенном рассуждении фиксировался номер паттерна k и уменьшали размерность N. Естественнее действовать наоборот: зафиксировать размерность N и, постепенно наращивая k, искать его максимальное значение, для которого уравнение (5.18) имеет решение. На рис. 5.6 показано поведение кривых  $F_k(y)$  из правой части уравнения (5.18) для различных значений k. Мы видим, что с ростом k критическая точка  $y_c^{(k)}$  тоже увеличивается. В то же время величина максимума  $F_k(y_c^{(k)})$  неуклонно понижается. Нетрудно отыскать максимальное значение k, для которого уравнение (24) имеет решение. Обозначим его через  $k_{\text{max}} = k_{\text{max}}(N)$ .

Для данной размерности *N* паттерн с номером  $k_{\text{max}}$  является последним, в окрестности которого существует локальный минимум функционала. Поскольку для  $k < k_{\text{max}}$  уравнение (5.18) тем более имеет решение в области  $y > y_c^{(k)}$ , в окрестности паттернов с меньшими номерами также будут существовать локальные минимумы. Иначе говоря, паттерны с номерами, меньшими  $k_{\text{max}}$ , сетью распознаются. Напротив, для  $k > k_{\text{max}}$  уравнение (5.18) не имеет решений в области  $y > y_c^{(k)}$ . Несмотря на то, что паттерны с такими номерами участвуют в образовании квази-Хеббовской матрицы, распознаваться они не будут. Обсудим этот феномен подробнее.

Хотя в матрице связи аккумулирована информация об очень большом числе паттернов (формально  $M = \infty$ ), «катастрофы памяти» не происходит. Память системы ограничена конечным числом паттернов, веса при которых превосходят критическое значение:  $r_{\mu} \ge r_c = 1/k_{\text{max}}$ . Паттерны с меньшими весами в памяти не помещаются. Стоит, однако, увеличить вес любого из таких паттернов, сделать его больше критического значения  $r_c$ , и паттерн будет распознаваться. Возможно, что при этом в разряд нераспознаваемых будут вытеснены какие-то другие паттерны, из числа тех, что раньше

распознавались. Такое замещение в «актуальной» памяти одних паттернов другими не только не противоречит здравому смыслу, но отвечает общим представлениям о свойствах человеческой памяти.

Таким образом, использование различных весов при паттернах: а) позволяют избежать катастрофы памяти, характерной для стандартной модели Хопфилда; б) наделяют память сети свойствами, которые имеют содержательную интерпретацию в психологических терминах.



**Рис. 5.6**. Поведение кривых  $F_k(y)$  из правой части уравнения (24) для k = 3, 6 и 9. Точка  $y_c^{(k)}$  с ростом k увеличивается, а максимум  $F_k(y_c^{(k)})$  понижается. Для заданного N легко отыскать максимальный номер паттерна  $k_{\text{max}}$  такой, что для  $k > k_{\text{max}}$  в области  $y > y_c^{(k)}$  уравнение (24) не имеет решения.

Обратимся теперь к критическому значению веса  $r_c = 1/k_{max}$ . В работах [58], [67] для  $r_c$  была получена оценка:  $r_c = \sqrt{2 \ln N/N}$ . Эта оценка получена с использованием статистической техники «сигнал/шум» при самых общих предположениях о распределении весов  $\{r_{\mu}\}_{\mu=1}^{\infty}$ . Предполагается только, что  $\sum_{\mu=1}^{\infty} r_{\mu}^2 = 1$ . Поскольку для обратного натурального ряда имеем  $\sum_{\mu=1}^{\infty} 1/\mu^2 = \pi^2/6$ , нормируем наши веса и приведем их к виду  $r_{\mu} = \sqrt{6}/(\pi\mu)$ . Следует ожидать, что  $\kappa_{max} = k_{max}(N)\pi/\sqrt{6}$  должно приблизительно равняться  $k_c(N)$ :

$$k_{c}(N) = \frac{1}{r_{c}} = \sqrt{\frac{N}{2\ln N}}.$$
(5.19)

На рис. 5.7 по оси абсцисс отложен  $\log_{10} N$  и показаны графики  $k_c(N)$ (штриховая линия) и  $\kappa_{max}(N)$  (сплошная линия). Мы видим, что в широком интервале изменения N значения этих двух характеристик близки друг к другу:  $\kappa_{max}(N) \approx k_c(N)$ . Данный результат интересен еще и потому, что  $\kappa_{max}$ найдено в результате решения уравнения (5.18), основанного на методах статистической физики, а выражение (5.19) для  $k_c$  получено методами, основанными на теоретико-вероятностном соотношении «сигнал/шум» [58], [67].



**Рис. 5.7**. Критические характеристики  $k_c$  (5.19) (штриховая линия) и  $\kappa_{\text{max}}$ , связанная с уравнением (5.18) (сплошная линия), и экспериментальные точки  $k_m^{\text{exp}}$  (показанные маркером), как функции  $\log_{10} N$  (см. текст).

На рис. 5.7 маркерами показаны экспериментально найденные критические номера паттернов для размерностей N=1000, 5000, 10000 и 25000. Для данного N генерировали случайную матрицу с весами в виде гармонического ряда и определяли максимальный номер паттерна, являющегося неподвижной точкой. В результате усреднения по 10 случайным матрицам получали  $k_m^{exp}$  - экспериментальную оценку для  $k_m$ . Видно, что эксперимент хорошо согласуется с теоретическими предсказаниями.

Зависящее от размерности N максимальное значение  $k_{\max}$ , фактически, играет ту же роль, что в стандартной модели Хопфилда отведена числу паттернов  $M_c = \alpha_c N$  - в обоих случаях речь идет о наибольшем числе

распознаваемых паттернов. Поскольку  $k_{\max}(N) \sim \sqrt{N / \ln N}$ , отношение  $k_{\max}(N) / N$  в асимптотике стремится к 0:

$$\lim_{N \to \infty} k_{\max}(N) / N \sim \lim_{N \to \infty} 1 / \sqrt{N \ln N} = 0$$
(5.20)

Иначе говоря, «емкость памяти» такой сети много меньше емкости памяти стандартной модели Хопфилда. По-видимому, это является следствием слишком быстрого убывания весов  $r_{\mu} = 1/\mu$ . Для весов, убывающих медленнее, число распознаваемых паттернов может оказаться пропорциональным N - см. следующий пункт.

5.46) Веса в виде геометрической прогрессии:  $r_{\mu} = q^{\mu}$ ,  $q \in (0,1)$ .

Для моделирования процессов старения памяти веса в виде убывающей геометрической прогрессии привлекались авторами [84], [85] (Nadal et al (1986), van Hemmen and Kuhn (1992)).

Начальное значение индекса суммирования естественно положить равным 0 и принять, что первый весовой множитель равен 1:  $r_0 = 1$ . Тогда имеем

$$f_{\mu}^{(k)}(y) \sim \frac{q^{2\mu}}{(q^k \varphi - q^{\mu})^2} = \left(\frac{q^{\mu}}{s - q^{\mu}}\right)^2$$
, где  $s_k = q^k \varphi(y)$  и  $\mu, k \ge 0$ .

Уравнение принимает вид:

$$N = \sum_{\mu=0\neq k}^{\infty} f_{\mu}^{(k)}(y) = \frac{1}{\gamma^2} \sum_{\mu=0\neq k}^{\infty} \left(\frac{q^{\mu}}{s_k - q^{\mu}}\right)^2.$$
 (5.21)

Нас интересует его решение при больших значениях аргумента, в области, где выполняется неравенство  $s_k = q^k \varphi(y) > 1$ .

При условии, что *q* близко к единице, в правой части уравнения (5.21) можно от суммирования перейти к интегрированию:

$$\sum_{\mu=0}^{M} \frac{q^{2\mu}}{(s_k - q^{\mu})^2} \approx M \int_0^1 \frac{q^{2Mx}}{(s_k - q^{Mx})^2} dx = \frac{\ln\left(\frac{s_k - q^M}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 - q^M)}{(s_k - 1)(s_k - q^M)}}{\ln q} \xrightarrow[M \to \infty]{} \frac{\ln\left(\frac{s_k}{s_k - 1}\right) - \frac{s_k(1 -$$

Вычитая отсюда член суммы с  $\mu = k$ , для правой части уравнения (5.21) окончательно получаем:

$$\sum_{\mu=0\neq k}^{\infty} f_{\mu}^{(k)}(y) = \frac{1}{\gamma^2} \left[ \Phi_k(y) - \frac{1}{(\varphi - 1)^2} \right], \quad \text{где} \quad \Phi_k(y) = \frac{\ln\left(\frac{s_k - 1}{s_k}\right) + \frac{1}{s_k - 1}}{\left|\ln q\right|} \quad \text{и}$$

 $s_k > 1$ .

Аналогом уравнения (5.18) теперь будет:

$$\frac{1}{N} = \frac{\gamma^2 (\varphi - 1)^2}{(\varphi - 1)^2 \Phi_k(y) - 1}.$$
(5.22)

Численно решая уравнение (5.22) для каждого q можно найти наибольший номер паттерна, который распознается сетью:  $k_m(N,q)$ . Нас интересует такое значение q, которому отвечает наибольшее число распознаваемых паттернов; обозначим это оптимальное q через  $q_m$ , а соответствующее число паттернов – через  $k_m(N)$ :

$$k_m(N) = k_m(N, q_m) = \max_q k_m(N, q).$$
(5.23)

Ясно, что оптимальное  $q_m$  должно существовать. В самом деле, пока q мало, число распознаваемых паттернов невелико: в этом случае веса  $r_{\mu} = q^{\mu}$  с ростом  $\mu$  убывают очень быстро, и распознаваться будут только самые первые паттерны (возможно, только один первый паттерн). Напротив, когда  $q \rightarrow 1$ , ситуация приближается к стандартной модели Хопфилда. При достаточно больших значениях q, превосходящих некоторое критическое  $q_c$ , ни один паттерн не будет распознаваться из-за «катастрофического забывания». Следовательно, для некоторого оптимального значения

 $q_m \in (0,1)$  число распознаваемых паттернов должно быть максимально:  $q_m = q_m(N)$ .

Легко отыскать критическое значение  $q_c$ , при котором распознается только самый первый паттерн - для  $q > q_c$  паттерны перестают распознаваться вообще. Можно показать, что для  $q_c$  справедлива оценка:

$$q_c = 1 - \delta$$
, где  $\delta \approx \frac{1}{0.329N}$ . (5.24)

Получить аналитическую оценку для оптимального значения  $q_m(N)$  пока не удается. Однако эту оценку можно получить, численно решая уравнение (5.22) для различных значений q и N.

На левой панели рис. 5.8, для трех размерностей N показана зависимость от q отношения  $k_{\max}(N,q)/N$ . Мы видим, что у всех кривых имеется отчетливо выраженный максимум. Для разных кривых максимум достигается при различных значениях  $q_m$ , однако величина максимума во всех случаях одна и та же:

$$\lim_{N \to \infty} \tilde{k}(N) / N = 0.05.$$
 (5.25)

Иначе говоря, максимальное число паттернов, которое способна хранить такая сеть, пропорционально размерности  $N: M_c \approx 0.05 \cdot N$ . Напомним, что для матрицы с весами в виде гармонического ряда асимптотика  $k_{\max}(N) / N$ равна 0 - см. (5.20). Такая разница в размере памяти обусловлено тем, что для начальных значений  $\mu$  веса  $r_{\mu} = 1 / \mu$  убывают гораздо быстрее, чем веса  $r_{\mu} = (q_m)^{\mu}$ , поскольку оптимальные значения  $q_m(N)$  все-таки достаточно близки к 1:  $q_m = 0.992, 0.9992$  и 0.99992 для N = 1000, 10000 и 100000 соответственно.

Если увеличивать параметр q за оптимальное значение  $q_m$ , число распознаваемых паттернов станет уменьшаться. Вскоре будет достигнуто критическое значение  $q_c$ , начиная с которого не будет распознаваться уже ни

один паттерн. Чем больше размерность N, тем  $q_c$  (и  $q_m$ ) ближе к 1. Для тех же что и выше размерностей N имеем:  $q_c = 0.997, 0.9997$  и 0.99997 соответственно.

На правой панели рис. 5.8 для паттерна с номером  $k_{\rm max}$  показана зависимость от q перекрытия с локальным минимумом. Интересно, что в точке «срыва» решения  $q_c$ , когда паттерны перестают распознаваться, величина перекрытия имеет приблизительно одно и то же значение,  $m_c \approx 0.933$ , и не зависит от размерности задачи.



**Рис. 5.8.** Для трех значений размерности N показаны: слева – зависимость от q отношения  $k_{\max} / N$ , где  $k_{\max}$  есть номер последнего распознаваемого паттерна; справа – синхронные значения перекрытия этого паттерна с ближайшим локальным минимумом. Сплошная линия - N=1000, штриховая линия - N=10000, точечная линия – N=100000 (см. текст). Точке максимума на левой панели отвечает  $\tilde{k}(N) / N$  - см. (5.25).

Все эти результаты получены численным решением уравнения (5.22). Можно продвинуться чуть дальше в аналитике, если для  $q \rightarrow 1$  упростить правую часть уравнения (5.22) для больших значений *y*:

$$\frac{1}{N} = g^2 \frac{2q^{2k} |\ln q|}{1 - 2q^{2k} |\ln q|},$$
(5.26)

где  $g = \varphi \cdot \gamma = \frac{erf y}{\sqrt{2}y} \approx \frac{1}{\sqrt{2}y}$ .



**Рис. 5.9.** Правая часть уравнения (5.22) (сплошная линия) и уравнения (5.26) штриховая линия для различных значений k и q: левая панель - q = 0.999, правая панель - q = 0.999. На обеих панелях нижние кривые отвечают k = 10, а верхние - k = 100.

На рис. 5.9 показано, как ведут себя правые части уравнения (5.22) и (5.26) при различных значениях k и q. Видно, что для больших значений аргумента  $y \ge 2$  правые части уравнений (5.22) и (5.26) практически совпадают.

Теперь уравнение (5.26) позволяет выписать явное выражение для k:

$$k = \frac{\ln \left[ 2(Ng^2 + 1) |\ln q| \right]}{2 |\ln q|}$$
(5.27)

Продифференцировав правую часть (5.27) по *q* и приравняв производную к нулю, получим, что при

$$q_0 \approx 1 - \frac{e}{2(Ng^2 + 1)}$$

величина *k* принимает максимальное значение:

$$k_0 \approx \frac{Ng^2 + 1}{e} \approx \frac{Ng^2}{e} \approx \frac{N}{2ey^2}$$

Потребуем, чтобы выполнялись условия идеального распознавания [82] (perfect recognition), когда паттерн и локальный минимум различаются меньше, чем на 1 бит. Это равносильно требованию: m = erf(y) > 1 - 1/N. Используя приближенное выражение для функции erf(y), получаем, что должно выполняться приближенное равенство:  $y \approx \sqrt{\ln(N/4)}$ . Тогда имеем:

$$\frac{k_0}{N} \approx \frac{1}{2e \ln(N/4)}$$
 (5.28)

Мы видим, что требование идеального распознавания приводит к сильным ограничениям на число паттернов. Действительно, если не требовать идеального распознавания и позволить паттерну отличаться от локального минимума на небольшую, но конечную величину, то число распознаваемых паттернов будет пропорционально N - см. (5.25). Если же потребовать выполнения условий идеального распознавания, то таких паттернов становится гораздо меньше: их число асимптотически стремится к нулю - см. (5.28). Заметим, что для стандартной модели Хопфилда хорошо известна оценка числа паттернов, которые распознаются идеально:  $M_0 / N \sim 1/2 \cdot \ln N$  (см., например, [82]). Мы видим, что, фактически, эта величина в *е* раз превосходит  $k_0 / N$ .

На рис. 5.10 показаны 3 графика, отвечающие размерности N=1000. Сплошной линией дана зависимость от q величины  $k_{\text{max}}$  (5.23); фактически, данная кривая повторяет сплошную линию на левой панели рис. 5.8 (без деления  $k_{\max}(q)$  на N=1000). Штриховой линией показана зависимость от qнаибольшего номера паттерна, для выполняется которого условие идеального распознавания. Будем обозначать эту характеристику  $k_{p}$  (от perfect). Она вычисляется подстановкой в (5.27) выражения  $y \approx \sqrt{\ln(N/4)}$ , являющегося условием идеального распознавания m > 1 - 1 / N. Наконец, маркером на рисунке показаны результаты компьютерного эксперимента, в определяли число паттернов, совпадающих с локальными котором минимумами. Результаты экспериментов усреднялись по 10 случайным матрицам. Мы видим в целом неплохое согласие теоретических результатов с компьютерным экспериментом.



**Рис. 5.10.** Для N = 1000 показана зависимость от q числа паттернов. Сплошная линия – число распознаваемых паттернов  $k_{\max}$ . Штриховая линия – число паттернов  $k_p$ , для которых происходит идеальное распознавание (см. текст). Обе эти кривые – теоретические. Маркеры – число идеально распознающихся паттернов  $k_p$ , найденное компьютерным моделированием.

5.4в) Веса в виде арифметической прогрессии:  $r_{\mu} = 1 - (\mu - 1)d$ ,  $\mu = 1, M$ .

Арифметическая прогрессия является идеализацией равномерно распределенных в интервале (0,1] случайных весов. Верхний предел суммирования в правой части уравнения (5.14) теперь должен быть конечным:  $M < \infty$ . Параметры M и N никак не связаны друг с другом: число паттернов M может быть как больше, так и меньше размерности задачи N. Еще один свободный параметр – шаг прогрессии d. Из того условия, что самый маленький член прогрессии  $r_M$  должен быть положительным, определяется максимально возможный шаг прогрессии  $d \le d_m$ , где

$$d_m = 1/M \iff r_M = 1/M > 0.$$

Внутри интервала  $(0, d_m]$  шаг прогрессии можно выбрать любым. Вначале мы проанализируем случай  $d = d_m$ , а зачем исследуем случай меньшего шага прогрессии  $d = d_m / g$ , где g > 1.

Преобразуем правую часть уравнения (5.14) подобно тому, как это делалось выше, именно, считая M очень большим, заменяем суммирование интегрированием, и получаем для правой части (5.14):

$$\sum_{\mu=0\neq k}^{M} f_{\mu}^{(k)}(y) = \frac{1}{\gamma^{2}} \left[ M \cdot \Phi_{k}(y) - \frac{1}{(\varphi - 1)^{2}} \right],$$

где

$$\Phi_k = \frac{2\varphi_k - 1}{\varphi_k - 1} + 2\varphi_k \ln\left(\frac{\varphi_k - 1}{\varphi_k}\right) \text{ and } \varphi_k = \varphi(y)\left(1 - \frac{k}{M}\right).$$
(5.29)

Вводя относительный номер паттерна  $\kappa = k / M$ , получаем аналог уравнений (5.18) и (5.22):

$$\alpha = \frac{\gamma^2}{\frac{2\varphi_k - 1}{\varphi_k - 1} + 2\varphi_k \ln\left(\frac{\varphi_k - 1}{\varphi_k}\right)}, \text{ где } \varphi_k = \varphi(y)(1 - \kappa).$$
(5.30)

Здесь  $\alpha$  - стандартное обозначение для параметра загрузки  $\alpha = M / N$ . Уравнение рассматривается в области больших значений *y*, там, где  $\varphi_k = \varphi(y)(1-\kappa) > 1$ . Нас будут интересовать решения этого уравнения, которые лежат правее максимума правой части (5.30).

На рис. 5.11 показано поведение правой части уравнения (5.30) для трех значений  $\kappa$ : 0, 0.49 и 0.6. Сплошная кривая отвечает  $\kappa = 0$  и характеризует условия распознавания самого первого паттерна, которому отвечает наибольший вес  $r_0 = 1$ . Мы видим, что максимальное значение параметра загрузки  $\alpha$ , при котором 1-й паттерн еще будет распознаваться, равно приблизительно  $\alpha(0) = 0.47$ . Для сетей с большей загрузкой  $\alpha > 0.47$  ни один паттерн распознаваться не будет. Штриховая кривая отвечает  $\kappa = k / M = 0.49$ и характеризует условия, при которых распознаются 49% из М паттернов, прописанных в матрицу связи. Для того, чтобы распознавалось столько паттернов, загрузка сети  $\alpha$  должна быть меньше 10%:  $\alpha(0.49) \approx 0.09$ . Если требуется распознавать 60% из М паттернов, параметр загрузки должен быть еще меньше (см. штрихпунктирную кривую):  $\alpha(0.6) \approx 0.05$  – и так далее. С увеличением  $\kappa$  максимальное значение параметра загрузки  $\alpha(\kappa)$  монотонно убывает. Это вполне объяснимо: чем большую часть паттернов к необходимо распознавать, тем больше должна быть размерность N и, следовательно, тем меньше будет значение параметра загрузки  $\alpha$ .



Рис. 5.11. Правая часть уравнения (5.30) для трёх различных значений  $\kappa$ .

Под емкостью памяти сети обычно понимают отношение числа распознаваемых паттернов k к размерности задачи N. Понятно, что  $k/N = \kappa \cdot \alpha$ . На рис. 5.12 показана зависимость от  $\kappa$  функций  $\alpha(\kappa)$  и  $\kappa \cdot \alpha$ . Для фиксированного  $\kappa$  величина  $\alpha(\kappa)$  определяется как максимум правой части уравнения (5.30). Как мы только что убедились, с ростом  $\kappa$  эти максимумы монотонно убывают (нижняя панель рис. 5.12). В то же время, кривая  $k/N = \kappa \cdot \alpha$  имеет четко выраженный максимум при  $\kappa_{\rm m} \approx 0.3$ , равный

$$k_{\rm m} / N \approx 0.06.$$
 (5.31)

Выражение (5.31) дает нам оптимальное значение емкости памяти сети с весами в виде арифметической прогрессии с максимально возможным шагом:  $d = d_m$ . Сравнив (5.31) с аналогичным выражением для геометрической прогрессии (5.25) мы видим, что веса в виде арифметической прогрессии позволяют получить большую емкость памяти, чем веса в виде геометрической прогрессии.



**Рис. 5.12**. Зависимость от  $\kappa$  максимальной загрузки сети  $\alpha(\kappa) = M / N$  и емкости памяти  $k / N = \kappa \alpha(\kappa)$ . Максимум емкости памяти достигается при  $\kappa \approx 0.3$  и равен  $k_m / N \approx 0.06$ .

Станем теперь уменьшать шаг арифметической прогрессии. Пусть  $d = d_m / g$  и g > 1. Раньше веса паттернов  $r_\mu$  убывали почти до нуля: самый маленький вес был очень малой величиной  $r_M = 1/M$ . Теперь веса распределены между 1 и b = 1 - 1/g. Легко проверяется, что уравнение (5.30) принимает вид:

$$\alpha = \frac{\gamma^2}{1 + \frac{\varphi_k^2}{(\varphi_k - 1)(\varphi_k - b)} + \frac{2\varphi_k}{1 - b} \ln\left(\frac{\varphi_k - 1}{\varphi_k - b}\right)}, \text{ где } \varphi_k = \varphi(y) \left(1 - \kappa / g\right). \quad (5.32)$$

Когда g = 1 правая часть уравнения (5.31) превращается в правую часть уравнения (5.30).

На рис. 5.13 показано, как в зависимости от параметра g меняется характер интересующей нас кривой  $\kappa \cdot \alpha(\kappa)$  (с верхней панели рис. 5.12). Максимум этой кривой дает оптимальное значение емкости памяти сети  $k_m / N$ . Мы видим, что с увеличением параметра g величина максимума возрастает, а точка максимума сдвигается вправо, в направлении  $\kappa_1 = 1$ . Начиная приблизительно с  $g \approx 4$  точка максимума кривой совпадает с ее правым концом  $\kappa_1$ . Дальнейшее увеличение g приводит только к «выпрямлению» кривой, постепенно превращающейся в прямую линию. При

неограниченно возрастающем *g* величина максимума на правом конце асимптотически стремится к емкости памяти стандартной модели Хопфилда:  $\alpha_c \approx 0.138$ .

Неограниченное увеличение *g* означает все более мелкий шаг прогрессии. Чем меньше шаг *d*, тем меньше различаются между собой наибольший  $r_0 = 1$  и наименьший  $r_M = 1 - 1/g$  веса. Тем больше вся ситуация приближается к стандартной модели Хопфилда. Отличие состоит в том, что теперь не происходит разрушения памяти вследствие ее переполнения.



**Рис. 5.13**. Емкость памяти  $k / N = \kappa \alpha(\kappa)$  для различных значений параметра *g*. С ростом *g* точка максимума кривой сдвигается к правому концу интервала.

#### 5.5 Выводы по главе 5

Мы показали обобщение модели Хопфилда на случай, когда паттерны, с Хеббовская матрица помощью которых строится связи, снабжены индивидуальными весовыми множителями  $\{r_{\mu}\}$ . Для самого общего вида распределения весов, методом седловой точки (в приближении репличной симметрии) получено основное уравнение, позволяющее отыскать минимум свободной энергии. Для общего случая исследованы свойства решения полученного уравнения. Для нескольких конкретных вариантов распределения весов решение уравнения исследовано подробно.

В качестве модели рассмотрена ситуация, когда только один вес  $r_1 = r$ отличается от всех остальных весов, тождественно равных 1:  $r_{\mu} \equiv 1$ ,  $\mu \ge 2$  и
*r*≠1. Также исследованы три частных случая различных весов: 1) Веса в виде гармонического ряда ( $r_k = 1/k$ ); 2) веса в виде затухающей геометрической прогрессии ( $r_k = q^k$ ) и 3) веса в виде убывающей арифметической прогрессии  $r_\mu = 1 - (\mu - 1)d$ .

Веса в виде гармонического ряда ( $r_k = 1/k$ ) убывают слишком быстро, поэтому максимальное число запомненных паттернов невелико по сравнению с размерностью задачи N. Как следует из формулы (5.19),  $k_m \approx \sqrt{N/2\ln N}$ .

Если веса взять в виде геометрической прогрессии  $r_k = q^k$ , то все будет определяться величиной параметра q. Было показано, что при некотором оптимальном значении q, достаточно близком к 1 ( $q_m \approx \sqrt{1-\delta_0}$ , где  $\delta_0 \approx 1/(0.165 \cdot N)$ ), число распознаваемых паттернов будет пропорционально N. Как следует из (5.25),  $k_m \approx 0.05N$ .

Для весов в виде арифметической прогрессии, при оптимальном значении параметра шага прогрессии, получается максимальное значение загрузки  $k_m \approx 0.06N$  (см.(5.31)).

Исследованные свойства для квазихеббовской модели качественно воспроизводят основные характеристики, присущие стандартной модели Хопфилда. Однако ни одним из рассмотренных способов распределения весов, не удалось превзойти ёмкость памяти (количество запоминаемых паттернов) чем в стандартной модели Хопфилда  $k_m \approx 0.14N$ .

В том случае, когда число паттернов, присутствующих в квазихеббовской матрице, бесконечно или больше числа запомненных  $k_m$ , то запоминаются только паттерны с большими весами, а с маленькими не запоминаются. На основании числа  $k_m$ , можно сделать оценку критического значения веса  $r_c$ , ниже которого паттерны не распознаются. В работах [58], [67] с использованием статистической техники «сигнал/шум», для  $r_c$  была

получена оценка:  $r_c = \sqrt{2 \ln N/N}$ . Эта оценка полностью подтвердилась для весов в виде гармонического ряда, поскольку оказывается, что  $r_c$  в точности равно  $1/k_m$ . При нормировке суммы квадратов весов на единицу ( $\sum r_m^2 = 1$ ), максимальное критическое значение веса  $r_c$  будет в случае одинаковых весов паттернов, т.е. как в стандартной модели Хопфилда. Таким образом, если статвес паттерна  $r_m$  больше критического значения для стандартной модели Хопфилда:  $r_c = 1/\sqrt{\alpha_c N}$ , где  $\alpha_c \approx 0.14$  - критическое значение параметра загрузки модели Хопфилда, то он точно будет минимумом.

## 6. ЗАКЛЮЧЕНИЕ

Настояшая диссертационная работа направлена на повышение эффективности процедуры случайного поиска, применяемой при решении задач бинарной минимизации. Эти задачи относятся к классу NP-сложных и характеризуются огромным (экспоненциальным от размерности задачи) числом локальных минимумов. Однако в главе 2 мы получили несколько результатов, которые показывают, что несмотря на большое количество минимумов, радиус области притяжения минимума напрямую зависит от его глубины: чем глубже минимум, тем больше его радиус области притяжения. Как следствие этого, вероятность отыскания растёт минимума экспоненциально с ростом его глубины, а большинство мелких локальных минимумов вообще оказываются не видимыми для случайного поиска.

Отталкиваясь от данных положений, наши усилия в решении задачи были направлены на трансформацию энергетической поверхности таким образом, чтобы увеличить глубину глобального минимума (всего на несколько процентов) и, тем самым, увеличить вероятность его отыскания (на несколько порядков).

В главе 3 мы рассмотрели простейший вид трансформации поверхности и предложили двухэтапный алгоритм минимизации DDK.

Подготовительная фаза алгоритма DDK состоит в следующем: 1) исходная матрица T симметризуется (если она изначально не симметрична) и диагональные матричные элементы обнуляются; 2) матрица возводится в k-ю степень и в полученной матрице  $M = T^k$  обнуляются диагональные элементы; 3) на основе матрицы M строится квадратичный функционал  $E_k(S)$ .

После подготовки можно приступать к процедуре случайного поиска на основе алгоритма двухэтапного спуска, состоящего в следующем: *a*) на первом этапе из начальной конфигурации осуществляется спуск по поверхности  $E_k(S)$  в ближайший локальный минимум  $S_m^{(k)}$  функционала

 $E_k(S)$ ; *b*) на втором этапе проводится коррекция – из точки  $S_m^{(k)}$  по поверхности  $E_1(S)$  спускаемся в ближайший локальный минимум  $S_m$  функционала  $E_1(S)$ , который как правило находится недалеко от точки  $S_m^{(k)}$ .

Анализ показывает, что алгоритм DD3 является оптимальным. Средняя эффективность алгоритма DD5 близка и даже лучше чем у DD3, однако DD5 менее стабилен. Например, для 2D EA матриц размера N = 100 алгоритм DD5 не находит глобальный минимум для 60% примеров, т.е. его эффективность очень высока для примерно 40% задач, и очень плоха для оставшихся 60%.

Добавление динамики Хоудайера-Мартина вместо нейросетевой динамики также значительно улучшает алгоритм. В частности, алгоритм DD3-HM позволяет найти глобальный минимум с вероятностью примерно 6%, которая в 30 раз выше чем вероятность (порядка 0.2%) даваемая алгоритмом DD3 и в 25000 раз больше чем вероятность ( $\Box 2.4 \cdot 10^{-6}$ ) даваемая алгоритмом SRS для 2D EA матриц размера N = 100.

Введение динамики Хоудайера-Мартина увеличивает время одного спуска примерно в десять раз (см. таблицу 3.5). Однако эта динамика добавляет стабильность в алгоритм: не было ни одной матрицы 2D EA размера N = 100, для которой DD3-HM алгоритм не нашёл бы глобальный минимум.

Для матриц двумерной модели Эдвардса-Андерсона размера N = 196, использование алгоритма DD3-HM позволяет добиться следующих улучшений: вероятность попадания в глобальный минимум увеличивается на порядок величины по сравнению с алгоритмом SRS-HM; разница между средней энергией  $\overline{E}_m$  и энергией глобального минимума уменьшается вдвое; вероятность попадания в интервал энергий близких к глобальному минимуму  $E_m \in [E_0, 0.99E_0]$  вырастает на порядок величины.

Также в главе 3 мы описали применение алгоритма DDK к задаче минимизации разреза графа на две равные части. На первом шаге

предлагается делать разбиение графа с матрицей связи  $T^k$ , где T - матрица связи исходного графа, k - некоторая степень (k = 2, 3, ..., 5). Полученное на первом шаге разбиение используется как начальное на втором шаге. Как показывают эксперименты, такой двухступенчатый алгоритм существенно повышает эффективность алгоритма разбиения графа.

А именно, эксперименты показывают, что, во-первых, такая трансформация связей графа понижает среднюю энергию находимых минимумов (разница между энергией глобального минимума и средней энергией находимых минимумов уменьшается почти в 3 раза, рис. 3.12). Как следствие, вероятность отыскания близких к глобальному минимумов повышается на порядки (для матриц Изинга уже при N = 100, а также для полносвязных матриц с равномерным распределением при больших размерностях начиная с N = 200).

Более того, с ростом размерности задачи, эффективность от использования возведения матрицы в степень, стремительно растёт.

Все эти результаты справедливы, какой бы ни был использован алгоритм локальной оптимизации разбиения графа: простейший жадный алгоритм (использованный в данной работе), алгоритм Кернигана-Лина, алгоритм имитации отжига или др.

В данной работе мы рассмотрели два типа графов, в которых матрицы связей имеют как положительные, так и отрицательные элементы. В практических задачах часто возникает потребность разбиения графов, рёбра которых неотрицательны. Дальнейшие исследования могут быть связаны с анализом эффективности предложенного алгоритма для такого рода матриц. Однако нами данный вопрос не исследовался.

В главе 4 был описан алгоритм минимизации MM, основанный на миксматрице (4.1).

Было показано, что использование микс-матрицы приводит к трансформации энергетической поверхности, в результате которой глубокие минимумы становятся существенно глубже, а мелкие – становятся еще

мельче или вовсе исчезают. В результате такой трансформации радиус любого глубокого области притяжения V минимума возрастает пропорционально его глубине. Как следствие, экспоненциально по N возрастает вероятность отыскания этого минимума. На этой основе предложен новый двухэтапный алгоритм случайного поиска ММ, который по эффективности и скорости отыскания глобального минимума значительно (экспоненциально по *N*) превышает качество работы стандартного алгоритма случайного поиска.

Следует отметить, что углубление минимума – это вероятностный процесс. Как следует из (4.3) с ростом параметра смешения z флуктуации величины  $E_{z0}$  /  $E_0$  увеличиваются (см. рис. 4.1). Это означает, что для какойто доли исследуемых матриц трансформация приведет к негативному результату – минимум станет мельче и вероятность его отыскания резко уменьшится. Действительно, такой эффект наблюдался с алгоритмом DD3, описанный в главе 3, который, несмотря на отличные усредненные характеристики, в случае матриц небольшой размерности (N~100) демонстрировал отсутствие стабильности при поиске глобального минимума: для 30% матриц вероятность отыскания глобального минимума снижалась до нуля из-за исчезновения на трансформированном функционале минимума вблизи S<sub>0</sub>. Алгоритм MM свободен от таких недостатков по двум причинам. Во-первых, уменьшение параметра z до значения  $z_{opt} \sim 0.5 - 0.7$  приводит к уменьшению флуктуации величины  $E_{z0} / E_0$  и, следовательно, вероятность негативного результата экспоненциально по N уменьшается (напомним, что дисперсия величины  $E_{z0}$  /  $E_0$  спадает как 1 / N). Во-вторых, сдвиг минимума в пространстве при  $z = z_{ont}$  в несколько раз меньше, чем при z = 1, что также положительно сказывается на стабильности и качестве работы алгоритма. Более того, при  $z = z_{ont}$  алгоритм демонстрирует не только высокую стабильность, но и самые высокие свои показатели.

Стандартный алгоритм локальной оптимизации SRS приводит к минимумам, отличающимся по энергии от глобального минимума на 10% для полносвязных матриц (SK) и на 17% для разреженных матриц типа матриц модели Изинга (2D и 3D EA). В то же время предложенный нами алгоритм MM приводит к нахождению минимумов, отличающихся по энергии от глобального минимума лишь на 4% для полносвязных матриц и на 6.5-7% для разреженных матриц (см. табл. 4.1).

Существенным аргументом в пользу алгоритма ММ является то, что с ростом размерности задачи средняя энергия находимых минимумов не меняется, а происходит лишь сужение дисперсии распределения находимых минимумов (см. рис. 4.5). Это означает, что алгоритм ММ даёт гарантированный выигрыш по энергии на 6.6% для полносвязных матриц и на 12% для разреженных матриц по сравнению с SRS. При этом время работы увеличивается менее чем в два раза, поскольку состоит из двух последовательных шагов, на каждом из которых выполняется локальный поиск (не считая затрат на возведение матрицы в квадрат, которое нужно произвести один раз вначале). Отличительной чертой алгоритма ММ является его простота.

ММ алгоритм может быть напрямую применён к задаче МАХСUТ (см. раздел 4.5) и показывает высокую конкурентоспособность по сравнению с другими известными алгоритмами для этой задачи. Сравнение с алгоритмами СirCut и Scatter Search показал, что алгоритм ММ даже за 10000 стартов (2с времени для задач с  $N = 8^3$  и 8с для задач с  $N = 15^3$ ) показывает результаты, сравнимые с одним из этих алгоритмов и не более чем на 3% хуже результата другого алгоритма, при том что CirCut и Scatter Search тратят на порядки большее время (см. табл. 4.2).

Ещё раз отметим, что ядром наших двухэтапных алгоритмов (как MM, так и DDK) является не модификация локального алгоритма поиска, а трансформация энергетической поверхности функционала. Это означает, что предложенные в работе алгоритмы можно значительно улучшить, заменив используемую здесь асинхронную динамику модели Хопфилда на более эффективную. Правда, как показывает практика, с ростом размерности любая динамика работает всё хуже и всё ближе приближается к динамике Хопфилда (как например динамика Хоудайера-Мартина в главе 3), в то время как разработанные в данной диссертации алгоритмы, основанные на трансформации поверхности, не ухудшают свои показатели при увеличении размерности задачи.

## СПИСОК ЛИТЕРАТУРЫ

- [1]. J. Houdayer, O.C. Martin. Hierarchical approach for computing spin glass ground states. Phys.Rev E, V. 64, 056704 (2001)
- [2]. Binder, K., Young, A.: Spin-glasses: Experimental facts, theoretical concepts and open questions. Rev.Mod.Phys. 58, 801 (1986)
- [3]. M'ezard, M., Parisi, G., Virasoro, M.A. Spin glass theory and beyond. World Scientific, Singapore(1987)
- [4]. Young, A.P. (ed.): Spin glasses and random fields. World Scientific, Singapore(1998)
- [5]. Nishimori, H.: Statistical Physics of Spin Glasses and Information Processing: An Introduction. Oxford University Press (2001)
- [6]. F. Barahona. On the computational complexity of Ising spin glass models. J. Phys. A: Math. Gen. 15(10), 3241 (1982) doi:10.1088/0305-4470/15/10/028
- [7]. Swendsen, R.H., Wang, J.S.: Replica Monte Carlo Simulation of Spin-Glasses. Phys.Rev.Lett. 57(21), 2607–2609(1986)
- [8]. Hartmann, A.K.: Cluster-exact approximation of spin glass ground states. Physica A 224,480–488 (1999)
- [9]. F. Belletti, M. Cotallo, A. Cruz, L.A. Fernandez, A. Gordillo, A. Maiorano, F. Mantovani, E. Marinari, V. Martin-Mayor, A. Munoz-Sudupe, D. Navarro, S. Perez-Gaviro, M. Rossi, J.J. Ruiz-Lorenzo, S.F. Schifano, D. Sciretti, A. Tarancon, J. L. Velasco "JANUS: an FPGA-based System for High Performance Scientific Computing". Computing in Science & Engineering, 11 (2009) 48-58.
- [10]. Ford, Jr., L.R., Fulkerson, D.R.: Maximal flow through a network. Canad. J. Math., 8, 399–404 (1956)
- [11]. A.K. Hartmann, H. Rieger. Optimization Algorithms in Physics. Wiley-VCH, Berlin (2001).
- [12]. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность. М.: Мир,1984.
- [13]. Гэри М., Джонсон Д.. Вычислительные машины и труднорешаемые задачи. М.: Мир. С. 416, 1982.
- [14]. Босс В. Лекции по математике. Т.10: Перебор и эффективные алгоритмы Учебное пособие. – М.: Издательство ЛКИ, 2008. – 216 с.
- [15]. Barahona, R. Maynard, R. Rammal, and J.P. Uhry, J. Phys. A 15, 673 (1982)
- [16]. M. Achilles, J. Bendisch, and H. v. Trotha, Physica A 275, 178 (2000)

- [17]. Shih, W.K., Wu, S., Kuo, Y.S.: Unifying maximum cut and minimum cut of a planar graph. IEEE Trans. Comput. 39(5) 694–697 (1990).
- [18]. F. Liers and G. Pardella. A Simple MAX-CUT Algorithm for Planar Graphs. In Proc. CTW, pp.351-354 (2009).
- [19]. L. Saul and M. Kardar. Exact Integer Algorithm for the 2D +/-J Ising Spin Glass.
  Phys.Rev. E 48, R3221 (1993); L. Saul and M. Kardar. The 2D +/-J Ising Spin Glass:
  Exact partition functions in polynomial time. Nucl.Phys. B 432, 641(1994).
- [20]. Buchheim, C., Liers, F., Oswald, M.: A basic toolbox for constrained quadratic 0/1 optimization. In: WEA. 249–262 (2008)
- [21]. Stoer, M., Wagner, F.: A simple min-cut algorithm. J. ACM 44(4) 585–591 (1997)
- [22]. A. Galluccio, M. Loebl, and J. Vondrak, Optimization via Enumeration: A New Algorithm for the Max Cut Problem, Mathematical Programming 90, 273(2001)
- [23]. F. Barahona. The Maxcut Problem on Graphs not Contractible to K5, Operations Research Letters 2, 107 (1983).
- [24]. McCormick, S.T., Rao, M.R., Rinaldi, G.: Easy and difficult objective functions for Max Cut. Math. Program. 94 (2-3, Ser.B) 459–466 The Aussois 2000 Workshop in Combinatorial Optimization (2003).
- [25]. Kolmogorov, V., Zabin, R.. What energy functions can be minimized via graph cuts?. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 147-159 (2004)
- [26]. J.C. Angles d'Auriac, M. Preissmann, and A.S. Leibniz-Imag, Math. Comp. Mod. 26, 1 (1997)
- [27]. H. Rieger, in J. Kertesz and I. Kondor (ed.): Advances in Computer Simulation, Lecture Notes in Physics 501, Springer, Heidelberg, 1998
- [28]. A. Hartwig, F. Daske, and S. Kobe, Comp. Phys. Commun. 32 133 (1984)
- [29]. T. Klotz and S. Kobe, J. Magn. Magn. Mat. 17, 1359 (1998)
- [30]. P. Stolorz, Phys. Rev. B 48, 3085 (1993)
- [31]. M.X. Goemans and D.P. Williamson. .878-approximation Algorithms for MAXCUT and MAX2SAT. ACM Symposium on Theory of Computing (STOC) (1994).
- [32]. M. Bellare, O. Goldreich and M. Sudan. Free Bits, PCPs and Nonapproximability. Towards Tight on Fundations of Computer Science, IEEE Computer Society 422-431 (1995)

- [33]. F. Liers, M. Junger, G. Reinelt, and G. Rinaldi. Computing Exact Ground States of Hard Ising Spin Glass Problems by Branch-and-Cut. Published in: New Optimization Algorithms in Physics. Wiley 2004, pp. 47-68 (2004)
- [34]. S.F. Edwards, P.W. Anderson, Theory of spin glasses, J. Phys. F: Met. Phys. 5, pp. 965–974 (1975)
- [35]. F. Rendl, G. Rinaldi, and A. Wiegele. Solving Max-Cut to Optimality by Intersecting Semidefinite and Polyhedral Relaxations. Math. Programming, v. 121, no. 2, p. 307 (2010).
- [36]. Wiegele A: Nonlinear Optimization Techniques Applied to Combinatorial Optimization Problems. Dissertation., Oktober 2006, i-x, pp. 1-131.
- [37]. Sherrington D., Kirkpatrick S. Solvable model of a spin-glass. Physical Review Letters, V. 35, P. 1972 (1975).
- [38]. Newman, M.E.J., Barkema, G.T.: Monte Carlo methods in statistical physics. Oxford University Press (1999)
- [39]. Kirkpatrick S., Gelatt C.D., and Vecchi M.P.. Optimization by Simulated Annealing. Science, vol. 220, No. 4598, pp. 671–680 (1983).
- [40]. D. Stauffer and P.M. Castro-de-Oliveira, Physica A 215, 407 (1995)
- [41]. P. Ocampo-Alfaro and Hong-Guo, Phys. Rev. E 53, 1982 (1996)
- [42]. B. A. Berg and T. Celik, Phys. Rev. Lett. 69, 2292 (1992)
- [43]. Geyer, C.: Markov Chain Monte Carlo Maximum Likelihood. In: Computing Science and Statistics, Proceedings of the 23<sup>rd</sup> Symposium on the Interface, pp.156–163. Interface Foundation of North America (1991).URL http://purl.umn.edu/58440
- [44]. E. Marinari and G. Parisi. Simulated Tempering: A New Monte Carlo Scheme. Europhys.Lett. 19(6), 451 (1992) doi:10.1209/0295-5075/19/6/002
- [45]. M. Yamashita, J. Phys. Soc. Jap. 64, 4083 (1995)
- [46]. P. Sutton and S. Boyden, Am. J. Phys. 62, 549 (1994)
- [47]. A. Pruegel-Bennett and S.L. Shapiro, Physica D 104, 75 (1997)
- [48]. K. Chen, Europhys. Lett. 43, 635 (1998)
- [49]. K.F. Pal, Physica A 223, 283 (1996); Physica A 233, 60 (1996)
- [50]. J. Houdayer and O.C. Martin, Renormalization for Discrete Optimization, Phys. Rev. Lett. 83(5) 1030-1033 (1999)
- [51]. J.J. Hopfield. Neural Networks and physical systems with emergent collective computational abilities. // Proc.Nat.Acad.Sci.USA. vol. 79, pp. 2554-2558 (1982).
- [52]. J.J. Hopfield, D.W. Tank. Neural computation of decisions in optimization problems. // Biological Cybernetics, vol.52, pp.141-152 (1985); J.J.Hopfield,

D.W.Tank. Computing with neural circuits: A Model. Science, vol. 233, pp. 625-633 (1986).

- [53]. Y. Fu, P.W. Anderson. Application of statistical mechanics to NP-complete problems in combinatorial optimization. // Journal of Physics A., vol.19, pp.1605-1620 (1986).
- [54]. T. Poggio, F. Girosi. Regularization algorithms for learning that are equivalent to multilayer networks. // Science 247, pp.978-982 (1990).
- [55]. S. Mulder and D. Wunsch II. A Million City Traveling Salesman Problem Solution by Divide and Conquer Clustering and Adaptive Resonance Neural Networks. // Neural Networks vol.16, No.5-6, pp.827-832 (2003).
- [56]. F. Wu and P.K.S. Tam. A neural network methodology of quadratic optimization.// International Journal of Neural Systems, vol. 9, No. 2 87-93 (1999).
- [57]. G. Pinkas, R. Dechter. Improving Connectionist Energy Minimization. // Journal of Artificial Inteligence Research, vol.3 (195), pp.23-48 (1995).
- [58]. Б.В. Крыжановский, Б.М.Магомедов, А.Л. Микаэлян. Взаимосвязь глубины локального минимума и вероятности его обнаружения в обобщенной модели Хопфилда. Доклады АН (2005) т. 405(3), с.320-324
- [59]. B. Kryzhanovsky, B. Magomedov. Application of domain neural network to optimization tasks. // Lecture Notes in Computere Science ICANN'2005. LNCS 3697, Part II, pp.397-403 (2005).
- [60]. New Optimization Algorithms in Physics. A.K. Hartmann and H. Rieger (Eds.), Wiley-VCH, Berlin (2004)
- [61]. W. Duch, J. Korczak. Optimization and global minimization methods suitable for neural networks. KMK UMK Technical Report 1/99; Neural Computing Surveys (1998). http://www.is.umk.pl/~duch/cv/papall.html
- [62]. L.B. Litinskii. Eigenvalue problem approach to discrete minimization. // W.Duch et al. (Eds.): ICANN 2005, LNCS 3697, pp. 405-410, 2005.
- [63]. K.A. Smith. Neural Networks for Combinatorial Optimization: A Review of More Than a Decade of Research. // INFORMS Journal on Computing v.11 (1), pp.15-34 (1999).
- [64]. G. Joya, M. Atencia and F. Sandoval. Hopfield Neural Networks for Optimization: Study of the Different Dynamics.// Neurocomputing, v.43(1-4), pp. 219-237 (2002).

- [65]. L.B. Litinskii, B.M. Magomedov. Global Minimization of a Quadratic Functional: Neural Networks Approach.// Pattern Recognition and Image Analysis v. 15(1), pp. 80-82 (2005).
- [66]. S. Boettecher. Extremal Optimization for Sherrington-Kirkpatrick Spin Glasses. // Eur. Phys. Journal B. 46, pp. 501 (2005).
- [67]. B.V. Kryzhanovsky, B.M. Magomedov, A.B. Fonarev. On the Probability of Finding Local Minima in Optimization Problems. // Proc. of Int. Joint Conf. on Neural Networks IJCNN-06, pp.5888-5892 (2006).
- [68]. B.V. Kryzhanovsky, V.M. Kryzhanovsky. The shape of a local minimum and the probability of its detection in random search. Lecture Notes in Electrical Engineering. Filipe, Joaquim; Ferrier, Jean-Louis; Andrade-Cetto, Juan (Eds.) Vol. 24, pp.51-61 (2009).
- [69]. Shneider, J.J., Dankesreiter, M., Fettes, W., Morgenstern, I., Schmid, M., Singer, J.M.. Search-space smoothing for combinatorial optimization problems. Physica A, 243(1), pp. 77-112 (1997)
- [70]. Zhang Y., Kihara D., Skolnick J.. Local energy landscape flattening: parallel hyperbolic Monte Carlo sampling of protein folding. Proteins 48(2),192-201 (2002)
- [71]. Pritchard-Bell, A., Shell, M.S.: Smoothing Protein Energy Landscapes by Integrating Folding Models with Structure Prediction. Biophys. J. 101(9), 2251–2259 (2011)
- [72]. B.V. Kryzhanovsky. Expansion of a matrix in terms of external products of configuration vectors. Optical Memory & Neural Networks (Information Optics) (2007) v. 16(4), pp.187-199
- [73]. D.J. Amit, H. Gutfreund, H. Sompolinsky. Spin-glass models of neural networks.
  Phys. Rev. A, vol.32, pp.1007-1018 (1985); Annals of Physics, vol.173, pp.30-67 (1987).
- [74]. B.W. Kernighan, S. Lin. An Efficient Heuristic Procedure for Partitioning Graphs// Bell System Tech. Journal, 1970, V. 49, P. 291-307.
- [75]. G. Karypis, V. Kumar. A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs. SIAM J. Sci. Comput., 1997, Vol. 20, No. 1, pp. 359-392.
- [76]. Hagen, L.; Kahng, A.B., "New spectral methods for ratio cut partitioning and clustering," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol.11, no.9, pp.1074,1085, Sep 1992
- [77]. http://www.optsicom.es/maxcut

- [78]. R. Marti , A. Duarte, M. Laguna. Advanced Scatter Search for the Max-Cut Problem. INFORMS Journal on Computing v.01, n.21, pp. 26-38 (2009).
- [79]. S. Burer, R.D.C. Monteiro, Y. Zhang. Rank-Two Relaxation Heuristics for Max-Cut and Other Binary Quadratic Programs. SIAM Journal on Optimization, v.12, pp.503-521 (2000)
- [80]. Литинский Л.Б. // Теоретическая и математическая физика, 1999, Т. 118 (1), С. 133-158.
- [81]. Amit D., Gutfreund H. and Sompolinsky H. // Phys. Rev. Letters, 1985, V. 55, P. 1530-1533.
- [82]. Hertz J., Krogh A., Palmer R.. Introduction to the Theory of Neural Computation. Addison-Wesley, NY, 1991.
- [83]. Г.Бейтмен, А.Эрдейи. Высшие трансцендентные функции, т.1, М.: Наука, 1965.
- [84]. J.P. Nadal, G. Toulouse, J.P. Changeux, S. Dehaene. Networks of Formal Neurons and Memory Palimpsets. Europhysics Letters, v.1 (10), pp. 535-542 (1986).
- [85]. J.L. van Hemmen, R. Kuhn. "Collective Phenomena in Neural Networks." In: "Models of Neural Networks", E. Domany, J.L van Hemmen and K. Shulten (Eds.), pp. 1-105, Berlin: Springer-Verlag, 1992.