

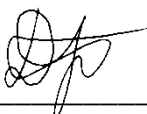
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Московский институт электроники и математики

Дюбайло Александр Вячеславович

МОДЕЛИРОВАНИЕ СПИНОВЫХ СТЕКОЛ

Выпускная квалификационная работа
студента образовательной программы
«Прикладная математика и информатика»
по направлению 01.03.02 Прикладная математика и информатика.



Студент

А. В. Дюбайло

Рецензент
ученое звание, должность

м.н.с. ИТФ им. Л.Д. Ландау, к. ф.-м. н.
Меньшутин А. Ю.

Руководитель ВКР
ученое звание, должность

профессор, д. ф.-м. н.
Щур Л. Н.

Москва 2016 г.

Аннотация

Предметом изучения настоящей выпускной квалификационной работы являются спиновые модели статистической физики на основе термодинамических величин двумерной модели Изинга на квадратной решетке. Для расчета основных параметров этой модели был рассмотрен алгоритм Метрополиса, для которого были произведены дополнительные оптимизации: AMSC, Parallel Tempering и др. В процессе исследования моделирование проводилось на программном комплексе, реализованном на языке C с использованием технологии программирования графических процессоров CUDA. Проведено сравнение результатов численного моделирования с результатами аналитического плана, полученных в прикладном пакете *Wolfram Mathematica* на основе статьи Пауля и Билле. Особое внимание уделено зависимости результатов моделирования от размера задачи. Результаты данной работы могут быть использованы при исследовании фазовых переходов в ферромагнетиках.

Abstract

The scopes of this study are spin models of the statistics physics. I consider the thermodynamic quantities of the two-dimensional Ising model on a square lattice. For calculations of the basic parameters of this model Metropolis algorithm was used with additional optimizations: AMSC, Parallel Tempering and others. The study was conducted on the simulation software package, implemented in C language using CUDA GPU programming technology. I compared numerical simulation results with the results of the analytical plan, obtained in the application package *Wolfram Mathematica* based on article of Paul and Bill. Particular attention was paid to the dependence of the simulation results on the problem size. The results of this work can be used in the study of phase transitions in ferromagnets.

Содержание

| | | |
|----------|------------------------------------------------------|-----------|
| 1 | Введение | 4 |
| 2 | Обзор имеющихся источников | 6 |
| 2.1 | Спиновое стекло | 10 |
| 3 | Двумерная модель Изинга. Алгоритм Метрополиса | 12 |
| 3.1 | Двумерная модель Изинга | 12 |
| 3.2 | Алгоритм Метрополиса | 13 |
| 3.3 | Оптимизация моделирования | 16 |
| 3.4 | Моделирование | 18 |
| 4 | Результаты моделирования | 22 |
| 4.1 | Анализ результатов | 25 |
| 4.2 | Эффекты размера модели | 29 |
| 5 | Выводы | 32 |
| | Список литературы | 33 |

1 Введение

Спиновые модели статистической физики позволяют описывать термодинамическое поведение многих веществ. Численное моделирование таких моделей может быть достаточно трудоемкой задачей. В данной работе особое внимание будет уделено двумерной модели Изинга. Актуальность этой выпускной работы обусловлена тем, что модель Изинга находится в центре разнообразных физических теорий, относящихся к критическим явлениям, жидкостям и растворам, спиновым стеклам, клеточным мембранам, моделированию иммунной системы, различным общественным явлениям и т.д. Кроме того, эта модель служит полигоном для проверки методов численного моделирования различных физических явлений.

Целью данной работы является изучение двумерной модели Изинга. Ставится необходимость разработки программного комплекса для численного моделирования двумерной модели Изинга с ближним взаимодействием спинов и получение ее основных термодинамических характеристик с использованием гибридного кластера. Также необходимо доказательство правильности проведенного моделирования.

Для достижения этой цели были поставлены следующие задачи:

- Разработка программного комплекса для моделирования этой модели на гибридном кластере
- Получение зависимостей основных термодинамических величин двумерной модели Изинга от температуры
- Анализ зависимостей и доказательство правильности моделирования для разных размеров модели
- Исследование влияния эффектов размера на полученные результаты моделирования

Так, в ходе данного исследования на языке программирования C с технологией CUDA была разработана программа для гибридного кластера с графическим ускорителем NVIDIA для получения основных зависимостей для двумерной модели Изинга. В результате моделирования были получены зависимости теплоемкости, внутренней энергии, намагниченности и магнитной восприимчивости для

такой модели. В этом исследовании был произведен анализ зависимостей, описанных выше, была доказана правильность моделирования. Особое внимание было уделено зависимости результатов моделирования от размера задачи.

Новизна исследования – характеризуется тем, что в дипломной работе реализован алгоритм Метрополиса с использованием декомпозиционной решетки типа шахматной доски для численного моделирования со следующими оптимизациями - упаковка спиновых значений, параллельное моделирование копий решетки при разных температурах и другие. Также произведен подробный анализ полученных результатов, построено точное решение модели и доказана правильность результатов моделирования.

Выпускная квалификационная работа состоит из введения, трёх глав, заключения и списка используемых источников.

В первой главе происходит описание используемых источников. Были рассмотрены несколько работ, связанных с этой тематикой, и произведен их анализ. Также были рассмотрены GPU системы и их описание, так как именно кластер с таким ускорителем был использован для моделирования. Во второй главе изложено подробное описание объекта исследования - двумерной модели Изинга. Описан алгоритм моделирования и его оптимизации. Так же рассказано о самом моделировании и приведены его параметры.

В третьей главе проведен анализ полученных результатов моделирования и их сравнение с аналитическим решением модели. В этой части работы приведены полученные зависимости. Также произведено сравнение времён моделирования с использованием гибридного кластера и только на CPU, показано ускорение работы за счет графического ускорителя.

Это исследования в будущем может быть использовано для изучения фазового перехода в ферромагнетиках. Результаты этой работы согласуются с аналитическим решением модели и могут служить для понимания природы ферромагнетизма. Алгоритмы и оптимизации к ним, реализованные в этой работе, могут быть применены к более сложным моделям статистической физики.

2 Обзор имеющихся источников

На данный момент существует множество работ, посвященных спиновым моделям статистической физики. В этой работе я хотел бы остановиться на статьях о модели Изинга, ее аналитическом решении и численному моделированию. Модель Изинга впервые была введена В. Ленцем в качестве задачи своему ученику Эрнсту Изингу. В своей работе [2] Изинг представил аналитическое решение одномерной модели и описал ее основную физику. Работа [3] посвящена аналитическому решению двумерной модели Изинга. Для аналитического решения модели Изинга применяется матричный метод. В 1942 году Л. Онсагер получил точное решение для квадратной решетки Изинга в нулевом магнитном поле. В 1949 году Онсагер и Кауфман рассчитали спиновые корреляционные функции для двумерной модели Изинга. В 1950-1955 гг. Ванье, Темперли, Хаутаппель, Сиози и др. рассмотрели треугольную, шестиугольную и более сложные типы двумерных решеток.

В работе [4] отражены основные положения двумерной модели Изинга на квадратной решетке с нулевым внешним магнитным полем и ее моделированию на компьютере. В данной работе получены зависимости, часть из которых будет получена мною и исследована. В этой работе получены зависимости внутренней энергии, намагниченности и других величин от температуры для моделей разных размеров.

Статья [5] посвящена алгоритму метрополиса, который используется для моделирования в этой работе. В этой работе представлены основные положения этого алгоритма и доказана его правильность, также показано, что данный алгоритм подходит для моделирования спиновых систем. Работа [1] посвящена моделированию Модели Изинга с использованием алгоритма метрополиса на GPU системах. В ней изложены основные способы декомпозиции решетки для моделирования на гибридных кластерах. Так она может быть положена в основу работ, посвященных моделированию спиновых систем на гибридных архитектурах.

В статьях [6], [12], [13] изложены различные оптимизации алгоритмов моделирования спиновых моделей на гибридных кластерах. Работа [13] содержит подробное описание технологии упаковки AMSC и представляет технологию моделирования *Parallel tempering*. За счет больших вычислительных возможностей, современные графические процессоры могут работать одновременно с множеством копий модели при разных температурах, что ускоряет моделирование. Работа [6] представляет некие технологии для оптимизации моделирования такие, как AMSC -

технология упаковки и рассказывает о декомпозициях решетчатых моделей для их моделирования на графических процессорах. В статье [12] рассказывается о эффективном моделировании спиновых систем с использованием векторных операций. Такая технология, как SMSC - упаковка спиновых переменных, может значительно ускорять моделирование на центральных процессорах. В этой дипломной работе она использоваться не будет, так как не является целью исследования.

Для моделирования многих спиновых систем необходимы случайные числа. Выбор генератора псевдослучайных чисел обусловлен множеством параметров: генератор должен обладать большим периодом и генерировать псевдослучайные числа независимые во многих потоках. В этой работе я использовал генераторы из библиотеки PRAND, описанные в статье [15]. Данная библиотека предоставляет широкий выбор генераторов псевдослучайных чисел, а в данной статье описаны правила их использования и основные параметры.

Так же в этом исследовании будет полезна работа [7]. В ней приведено аналитическое решение двумерной модели Изинга определенного размера с периодическими граничными условиями с нулевым внешним магнитным полем. Имеено оно будет использовано для проверки результатов моделирования. В этой работе изложено построение статистической суммы для двумерной модели Изинга конечного размера. Зная статистическую сумму, из нее можно найти остальные термодинамические величины модели. Такое решение было построено в прикладном пакете *Wolfram Mathematica*.

Описание GPU систем подробно изложено в книгах [10] и [11]. В них описаны все подробности архитектуры GPU и особенности, которые необходимо учитывать, при написании программ. Так же в этих книгах изложено подробное описание расширения языка C для программирования графических процессоров - CUDA. Именно язык C с технологией CUDA использовался для моделирования.

В этой работе для расчетов основных термодинамических параметров спиновой модели использовался кластер с гибридной архитектурой. Для эффективного использования таких кластеров необходимо учитывать особенности архитектуры, дабы оптимизировать расчеты, взаимодействие между устройствами и доступ к памяти.

На рисунке 1 показано схематичное представление NVIDIA GPU, который был использован в этой работе. GPU (graphics processing unit) или графический процессор отдельное устройство персонального компьютера или игровой приставки,

выполняющее работу с графикой. С середины 2000-х годов GPU стали массово применяться и в других устройствах: планшетные компьютеры, встраиваемые системы, цифровые телевизоры. Современные GPU очень эффективно обрабатывают и отображают компьютерную графику. Благодаря специализированной конвейерной архитектуре они намного эффективнее в обработке графической информации, чем типичный центральный процессор. Графический процессор в современных видеоадаптерах применяется в качестве ускорителя трёхмерной графики.

Современный GPU состоит из некоторого числа мультипроцессоров, каждый из которых содержит некоторое число процессорных элементов, которые работают со своими входными данными. На графическом процессоре имеется несколько видов памяти, которые позволяют достигнуть хорошей эффективности по времени при выполнении программ.

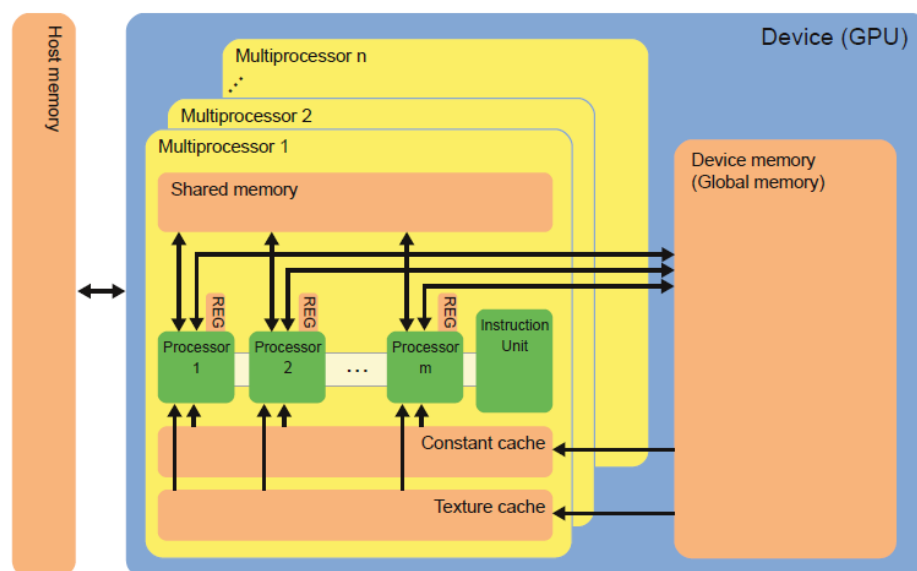


Рис. 1: Графическое представление GPU (взято из [1])

На современных графических процессорах имеются следующие типы памяти:

- *Регистровая память*: в каждом мультипроцессоре имеется несколько тысяч регистров, доступ к ним возможен только из этого мультипроцессора и осуществляется очень быстро (порядка 2 - 3 тактов)[10]
- *Разделяемая память*: все процессы, обрабатываемые на мультипроцессоре, имеют доступ к общей небольшой (от 16 до 48 кб) памяти, которая разделена

между всеми процессами мультипроцессора. Она служит для синхронизации нитей в одном блоке и доступ к ней может осуществляться за 5 - 6 тактов.

- *Глобальная память*: Основная память на графическом процессоре, ее объем сейчас достигает 12 GB. По скорости работы она самая медленная, доступ к ней осуществляется за 400 - 600 тактов.
- *Текстурная и Константная память*: Эти виды памяти доступны только для чтения и служат для хранения и обработки констант. По скорости эти виды памяти чуть уступают разделяемой памяти.

Разработку графических процессоров для вычислений компания NVIDIA начала в 2008 году. На данный момент выпущены три архитектуры графических процессоров. В 2012 году была представлена последняя архитектура GPU: Kepler. На этой архитектуре появилась возможность выбирать размер разделяемой памяти (16, 32 или 48 KB). А также появился еще один уровень Кэш - памяти, которая служит для кэширования переменных, которые не попадают в регистры. Все это позволяет оптимизировать доступ к памяти и тем самым ускорить выполнение программы.

Для наиболее эффективной реализации программ необходимо учитывать все эти особенности GPU. Так можно выделить несколько ключевых идей для эффективного использования графических процессоров:[11]

1. Расчеты должны обладать высокой степенью локализации, чтобы снизить необходимость взаимодействия нитей.
2. Наибольшая часть вычислительных ресурсов уходит на доступ к глобальной памяти, а не на арифметические операции.
3. Количество запущенных нитей может превышать доступное число процессорных элементов .

Именно такие особенности позволяют значительно ускорить выполнения некоторого ряда программ с использованием кластеров, основанных на гибридной архитектуре.

2.1 Спиновое стекло

Трехмерное спиновое стекло - одна из моделей статистической физики, заданная на трехмерной кубической решетке со следующим Гамильтонианом:

$$H = - \sum_{\langle ik \rangle} J_{ik} \sigma_i \sigma_k \quad (1)$$

где $\sigma_i \in \{-1; +1\}$ - значение спинов, J_{ik} - так называемые константы взаимодействия. В общем случае они могут быть любыми, принадлежать любому распределению. В зависимости от их значений, модель принадлежит тому или иному классу. Сумма берется по ближайшим соседям, в трехмерном случае их 6. Схематичное представление такой модели изображено на рисунке 2. В таких системах

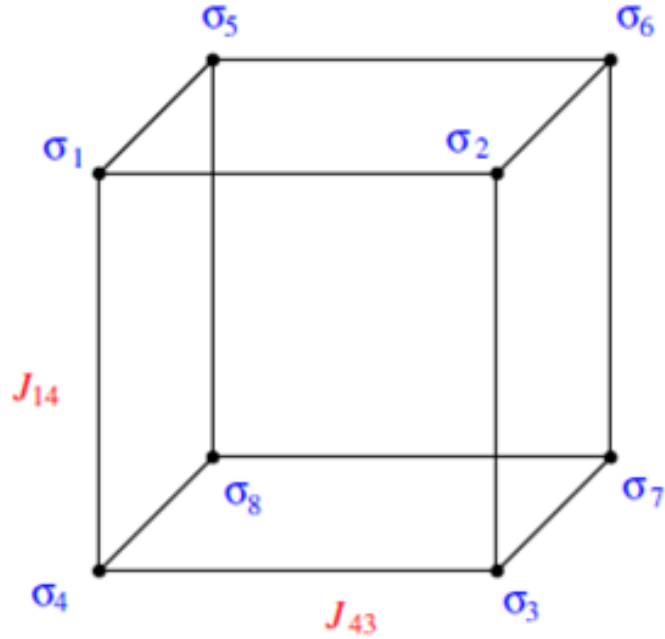


Рис. 2: Схематичное представление модели

имеется фазовый переход при критической температуре $T_C = 1.1019(29)$. При изучении таких систем, главный объект исследования - это нахождение равновесного состояния (то есть наиболее вероятного в природе). Вероятность нахождения такой системы в определенном состоянии $\{\sigma_i\}$ при определенной температуре β :

$$P(\{\sigma_i\}, \beta) = \frac{e^{-\beta H[\{\sigma_i\}]}}{\sum_{\{\sigma_i\}} e^{-\beta H[\{\sigma_i\}]}} \quad (2)$$

где $\beta = T^{-1}$. Основное состояние - наиболее вероятное, оно определяется минимумом энергии. Нахождение такого состояния сводится к минимизации гамильтониана (1), т. е. нахождению глобального минимума из всех возможных значений.

Обычно, для численного моделирования спиновых стекол используется метод Монте-Карло. Рассматривается трехмерная кубическая решетка с периодическими граничными условиями. Локальная энергия k -го спина в трехмерной модели, линейного размера L :

$$E(\sigma_k) = -\sigma_k \sum_{m(k)} J_{km} \sigma_m \quad (3)$$

где сумма берется по 6 соседям k -го спина. Вероятность значения спина $+1$ или -1 определяется согласно Heat-Bath алгоритму: [12]

$$P(\sigma_k = +1) = \frac{e^{\phi_k/T}}{e^{\phi_k/T} + e^{-\phi_k/T}} \quad (4)$$

$$\phi_k = \sum_{m(k)} J_{km} \sigma_m \quad (5)$$

Полный алгоритм может быть представлен следующими шагами:

1. Выбирается случайным образом k -ый спин
2. Происходить расчет локального поля по формуле (5)
3. $\sigma_k = +1$ с вероятностью $P(\sigma_k = +1)$. Для этого создается случайное число равномерно распределенное на отрезке $[0; 1] : r \in R[0; 1]$. If($r < P(\sigma_k = +1)$) $\sigma_k = 1$; else $\sigma_k = -1$.
4. Переход к шагу 1.

L^3 таких проходов называется одним Монте-Карло шагом (1 MCS). При большом количестве таких шагов система может достигнуть термодинамического равновесия, т. е. можно будет говорить о минимизации Гамильтониана (1) [12] [13].

3 Двумерная модель Изинга. Алгоритм Метрополиса

Основной объект исследования в этой работе - Двумерная модель Изинга на квадратной решетке. Модель Изинга — математическая модель статистической физики, предназначенная для описания ферромагнетика. Модель Изинга был введена физиком Вильгельмом Ленцом (1920) в качестве задачи своему ученику Эрнсту Изингу. Одномерная модель Изинга не имеет фазовый переход и была решена Изингом в 1924 году в работе [2]. Двумерная модель Изинга была аналитически решена Ларсом Онсагером в работе [3].

В основе многих физических явлений положенна двумерная модель Изинга, большая часть алгоритмов в статистической физики проверяется также на этой модели, так как в некоторых случаях она имеет аналитическое решение и следовательно правильность моделирования может быть проверена. В этой работе я использовал алгоритм разложения шахматной доски - оптимизация алгоритма Метрополиса для использования на гибридных кластерах для численного моделирования. Такое разложение может быть применено к другим моделям, где имеются локальные взаимодействия спинов.[14] Поэтому дальнейшая часть работы будет посвящена алгоритму Метрополиса с использованием GPU систем и его применению к модели Изинга на квадратной решетке.

3.1 Двумерная модель Изинга

Я исследовал двумерную модель Изинга на квадратной решетке размером $L \times L$. Каждой из 2^N возможных конфигураций спинов (где $N = L^2$) сопоставляется энергия взаимодействия соседствующих спинов:

$$E(\sigma) = -J \sum_{i \sim j} \sigma_i \sigma_j \quad (6)$$

Полная энергия такой системы, может быть записана гамильтонианом вида (7).

$$H(\sigma) = -J \sum_{i \sim j} \sigma_i \sigma_j - h \sum_i \sigma_i \quad (7)$$

Где σ - определенная конфигурация спинов, J - энергия взаимодействия, h - ве-

личина внешнего магнитного поля, $\sigma_{ij} = \pm 1$ - спины в узлах решетки.

Для моделирования использовалась такая модель Изинга с нулевым внешним магнитном полем и периодическими граничными условиями. Гамильтониан такой системы имеет следующий вид:

$$H(\sigma) = - \sum_{i \sim j} \sigma_i \sigma_j \quad (8)$$

Основной задачей, которая ставится в работах с такими моделями, является задача нахождения основного состояния данной системы или состояния термодинамического равновесия. Так система размером $N = L^2$ имеет 2^N состояний. Вероятность нахождения системы в определенном состоянии с конфигурацией спинов σ при определенной температуре задана формулой [4]:

$$P_\beta(\sigma) = \frac{e^{-\beta H(\sigma)}}{Z_\beta} \quad (9)$$

В формуле (9) $Z_\beta = \sum_{\sigma} e^{-\beta H(\sigma)}$ - сумма по всем конфигурациям (стат-сумма), $\beta = 1/k_b T$ - обратная температура, σ - определенная конфигурация спинов. Основное состояние - наиболее вероятное, оно определяется минимумом энергии. Нахождение такого состояния сводится к минимизации гамильтониана (8), т. е. нахождению глобального минимума из 2^N возможных значений.

Данная модель при размерности больше единицы имеет фазовый переход первого рода. Температура перехода, или точка Кюри для двумерной модели Изинга: $T_c = 2.269[J/k_b]$ [14].

3.2 Алгоритм Метрополиса

Приведенный в этой работе алгоритм был описан в статье [1]. Он основан на алгоритмах метрополиса [5] и модернизирован для использования на GPU системах.

Я использовал модель Изинга, описанную выше с гамильтонианом вида (8). Так же известна вероятность переворота спина [1] (вероятность перехода по Метрополису):

$$p_{acc}(s_i \mapsto -s_i) = \min[1, e^{-\beta \Delta E}] \quad (10)$$

Где $\beta = \frac{1}{k_b T}$ - обратная температура, k_b - константа Больцмана, ΔE - разница ло-

кальных энергий взаимодействия. В двумерной модели Изинга рассматриваются взаимодействия с 4 соседними спинами и вычисляется по формуле:

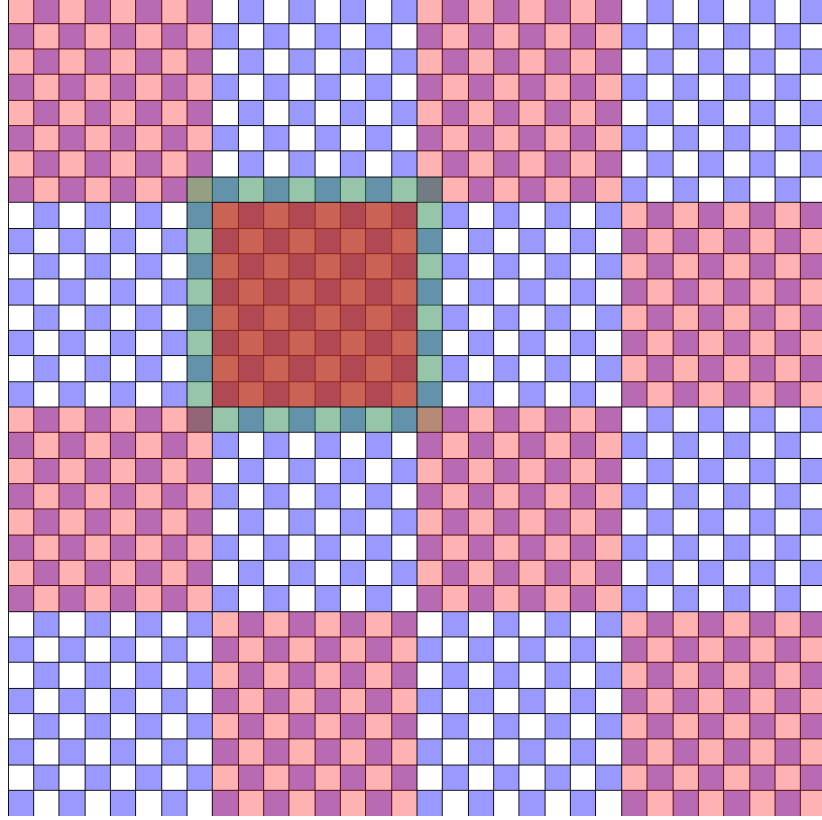


Рис. 3: Двумерная шахматная доска разложения для решетки размером 32×32 (взято из [1])

Обновление спинов в простейшем алгоритме происходит так:

1. Выбирается случайный спин на решетке.
2. Вычисляется энергия взаимодействия спина с соседями по формуле (11).

$$E_i = -\sigma_i \sum_{j=1}^4 \sigma_j \quad (11)$$

3. Значение спина меняется на противоположное и вычисляется разница энергий до переворота спина и после него. Если разница меньше нуля, то значение спина меняется на противоположное, так как система стремится к минимуму энергии.
4. Если разница больше нуля, то вероятность перехода спина из одного состояния в другое задается формулой (10). Создается псевдослучайное число $r \sim R[0, 1]$. Если $r < e^{-\beta \Delta E}$, то значение спина меняется.

Данный алгоритм можно распараллелить при помощи двойной шахматной доски. Так квадратная решетка подвергается двум уровням шахматного разложения. На верхнем уровне, вся решетка разбивается на блоки размером $B \times B$. На нижнем уровне, каждый блок разбивается методом шахматной доски на блоки размером $T \times T$. Это показано на рисунке 3.

В результате такого разложения все четные и нечетные блоки второго уровня разложения в каждой подрешетки первого уровня разложения могут обрабатываться параллельно. Верхний уровень разбиения необходим для разделения решетки между мультипроцессорами GPU. Так один мультипроцессор может одновременно обрабатывать 1024 нити и каждая нить графического процесса обрабатывает свой спин, поэтому для решеток, размером больше 32×32 необходим верхний уровень разделения.

Нижний уровень разбиения необходим для того, чтобы при расчете взаимодействия спина с его соседями, соседи не обновлялись, дабы вычисления были правильными. Так в блоки размером B делятся на отдельные блоки по одному спину и в параллель могут одновременно обрабатываться $B/2$ четных или нечетных спинов в каждом блоке. То есть при таком разбиении каждая нить считает взаимодействия одного спина с его соседями и параллельно обрабатываются все четные или нечетные спины, что и ускоряет само моделирование.

Алгоритм обновления спинов:

1. Cuda-ядро запускается для всех $B^2/2$ *нечетных* подрешеток первого уровня разбиения для распределения решетки по блокам, которые будут выполняться одновременно на мультипроцессорах.
2. $B/2$ нитей каждого блока подвергают процедуре обновления по Метрополису(описано выше) все *нечетные* подрешетки второго уровня разложения.
3. Происходит синхронизация всех нитей, чтобы убедиться, что процедура обновления завершена для всех нитей.
4. $B/2$ нитей каждого блока подвергают процедуре обновления по Метрополису все *четные* подрешетки второго уровня разложения по тому же алгоритму.
5. Происходит синхронизация в каждом блоке.
6. Функция ядра запускается для всех $B^2/2$ *четных* подрешеток первого уровня разбиения для распределения решетки по блокам, которые будут выпол-

няться одновременно на мультипроцессорах. Процедура обновления спинов происходит таким же образом.

Значительное ускорение моделирования происходит за счет того, что параллельно обновляется $L^2/2$ спинов решетки. Одно повторение шагов 1 - 5 называется один Монте-Карло шаг (1 MCS). На практике необходимо совершить количество MCS для данной модели порядка $10^5 \sim 10^6$ для достижения термодинамического равновесия. [9]

3.3 Оптимизация моделирования

Можно дополнительно оптимизировать и ускорить алгоритм. Для обновления спинов необходимо вычисление больцмановских экспонент в формуле (10). Так как значения спина $+1$ или -1 , то значение локальной энергии в уравнении (11) может принимать всего 10 значений: 2 значения для самого спина и 5 значений для суммы по соседям. То есть всего возможно 10 значений экспонент, которые можно вычислить заранее и загрузить в одну из быстрых памятей GPU - константную или текстурную, и использовать эти значения.

Для уменьшения времени доступа к памяти и работы с ней, все подрешетки вернего уровня разложения можно загрузить в разделяемую память GPU. Тогда значительно ускорится доступ к значениям спинов в подрешетках и сократится доступ к глобальной памяти GPU [1]. То есть теперь вся подрешетка первого уровня разложения будет загружена в разделяемую память, уже непосредственно в этой памяти спины будут подвергаться обновлению. Так необходимо всего два доступа к глобальной памяти: при выгрузке подрешетки в разделяемую память и при загрузке решетки в глобальную память.

Расчет термодинамических макро параметров для разных температур может также проходить в параллель [13]. GPU может работать с большим количеством нитей. Поэтому одновременно можно работать с множеством копий решетки, каждая из которых имеет свою температуру.

Значение спинов в этой модели принимает всего два значения: $+1$ и -1 . Поэтому целесообразно для хранения этих значений использовать бит одного числа, а не байты разных чисел. Для этого значения спинов нужно преобразовать следующим образом:

$$\sigma_i \rightarrow s_i, s_i = \frac{1 - \sigma_i}{2} \quad (12)$$

Такая процедура позволит "упаковать" значения спинов в биты M -битных слов. Я использовал упаковку спинов называемую Asynchronous multispin-coding (AMSC)[6]. Ее основная идея: в разные биты одного слова упаковываются значения спинов из разных копий решетки, но с одними и теми же координатами. Я работал с M копиями решетки S размером L^2 , для упаковки необходимо L^2 M -битных слов. Для самой упаковки использовались операции с битами: сдвиг влево ($<<$) и логическое ИЛИ ($|$).

Для упаковки j -го спина i -ой копии решетки ($S_i[j]$) в i -ю позицию j -го слова массива с упакованными спинами (B) нужно сдвинуть влево значимый бит j -го спина на i бит, затем сшить два слова операцией ИЛИ. Имеем

| | | | | | | |
|--------|------|----------|---|---|--------|--------|
| $B[j]$ | | i 0 | 0 | 0 | 1 0 | 0 |
| $S[j]$ | | 0 | 0 | 0 | 0 | 0 1 |

Сдвигаем влево на i -ю позицию значимый бит j -го элемента получаем:

| | | | | | | |
|--------|------|----------|---|---|--------|--------|
| $B[j]$ | | i 0 | 0 | 0 | 1 0 | 0 |
| $S[j]$ | | 0 | 1 | 0 | 0 | 0 0 |

Логическое ИЛИ обновляет i -й бит в j -слове:

| | | | | | | |
|--------|------|----------|---|---|--------|--------|
| $B[j]$ | | i 0 | 1 | 0 | 1 0 | 0 |
| $S[j]$ | | 0 | 1 | 0 | 0 | 0 0 |

Для обновления спинов и проведения моделирования необходима их распаковка. Для использования значения спина s в i -ой позиции цепочки *value* необходимо

преобразовать его к значению V по следующей формуле:

$$V = 1 - 2 \cdot ((value \gg i) \& 1) \quad (13)$$

Операция \gg - сдвиг вправо, $\&$ - побитовое И.

Обновление происходит не одного значения спина, а всей цепочки упакованных спинов одной операцией с помощью маски ($mask$). Маска инициализируется нулевым значением ($mask = 0$), далее при необходимости обновления i -го спина - в i -ый бит маски записывается единичное значение. Это делается так:

$$mask = mask | (1 \ll i) \quad (14)$$

После полного составления маски, обновление всех спинов из упакованной цепочки $value$ происходит одной операцией с помощью исключающего ИЛИ (\wedge):

$$value = value \wedge mask; \quad (15)$$

Данные оптимизации достаточно просты и помогают ускорить моделирование, особенно для решеток большого размера. Так же упаковка спинов экономит ресурсы GPU и сокращает количество арифметических операций. Так же дополнительное ускорение получается за счет уменьшения числа генерируемых псевдослучайных чисел и передачи информации между CPU и GPU.

3.4 Моделирование

Я считал намагниченность, магнитную восприимчивость, внутреннюю энергию и теплоемкость модели Изинга (8). Моделирование происходило на графическом процессоре NVIDIA Tesla K40. Были рассмотрены решетки с линейными размерами L равными 8, 16, 32, 48, 64, 96, 128 и 256. Я проводил моделирование как с упаковкой спинов AMSC, так и без нее. Упаковка происходила в 64-битные слова ($M = 64$). Количество Монте-Карло шагов для достижения термодинамического равновесия: $MCS_1 = 10^5$. Количество Монте-Карло шагов для расчета термодинамических параметров: $MCS_2 = 10^5$. Генерация псевдослучайных чисел происходила в каждом параллельном потоке независимо. Я использовал генератор псевдослучайных чисел MT19937 с периодом $4.3 \cdot 10^{6001}$ из библиотеки PRAND [15].

Намагниченность вычислялась по формуле:

$$M = \frac{\sum_{i=1}^{L^2} s_i}{L^2} \quad (16)$$

Магнитную восприимчивость я считал по формуле:

$$\chi = \frac{\langle M^2 \rangle - \langle M \rangle^2}{L \cdot T} \quad (17)$$

Так как в модели Изинга в данной работе взаимодействуют только соседние спины, то энергия взаимодействия одного спина с 4 соседями вычисляется по формуле:

$$E_1 = -s_1 \sum_i s_i \quad (18)$$

Тогда энергию всей системы вычисляется как сумму энергий взаимодействия всех спинов с соседями по формуле:

$$E_{full} = \frac{\sum_{i=1}^{L^2} E_i}{L^2} \quad (19)$$

Теплоемкость вычисляется как флуктуации энергии по следующей формуле:

$$C = \frac{\langle E^2 \rangle - \langle E \rangle^2}{L^2 \cdot T^2} \quad (20)$$

Для вычисления статистических погрешностей было выполнено 200 независимых запусков моделирования, на каждом запуске считались намагниченность, магнитная восприимчивость, внутренняя энергия и теплоемкость. Было подсчитано стандартное отклонение среднего значения величин по формуле:

$$s_{\tilde{x}_i} = \sqrt{\sum_{i=1}^n (x_i - \tilde{x})^2 / [n \cdot (n - 1)]} \quad (21)$$

Сценарий программы для одного независимого запуска с использованием всех оптимизаций, описанных выше можно увидеть на рисунке 4.

Ядро для обновления спинов запускается для двумерной квадратной сетке ни-

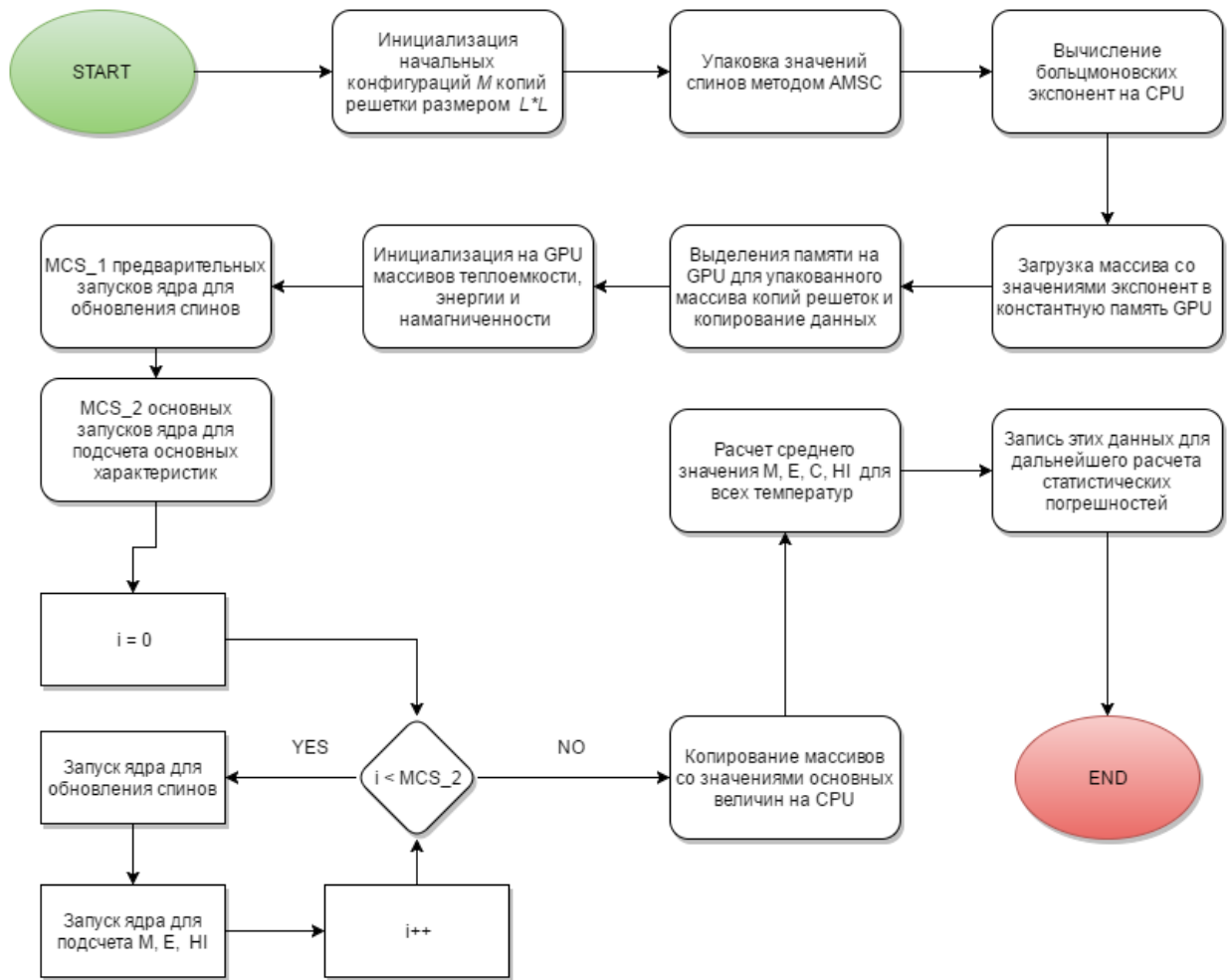


Рис. 4: Блок схема моделирования

тей, поделенных на блоки. Линейный размер сетки - L . Так конкретная нить ядра работает с определенной цепочкой упакованных спинов. Весь массив упакованных спинов разбивается на блоки по одной цепочки спинов методом двойной шахматной доски и каждая нить работает с соответствующим по координатам блоком. Так, если некоторая цепочка спинов имеет координаты (i, j) на двумерной решетке, то ее обрабатывает нить с такими же глобальными координатами в сетке нитей. Схема ядра для конкретной нити представлена на рисунке 5.

Данное ядро сначала обновляет все четные, затем все нечетные цепочки упакованных спинов. Одно случайное число и одна нить необходима для обновления M спинов с одинаковыми координатами из каждой копии решетки. Так же это ядро рассчитывает локальную энергию взаимодействия для каждого спина каждой копии решетки, которая в дальнейшем необходима для расчета внутренней энергии и теплоемкости.

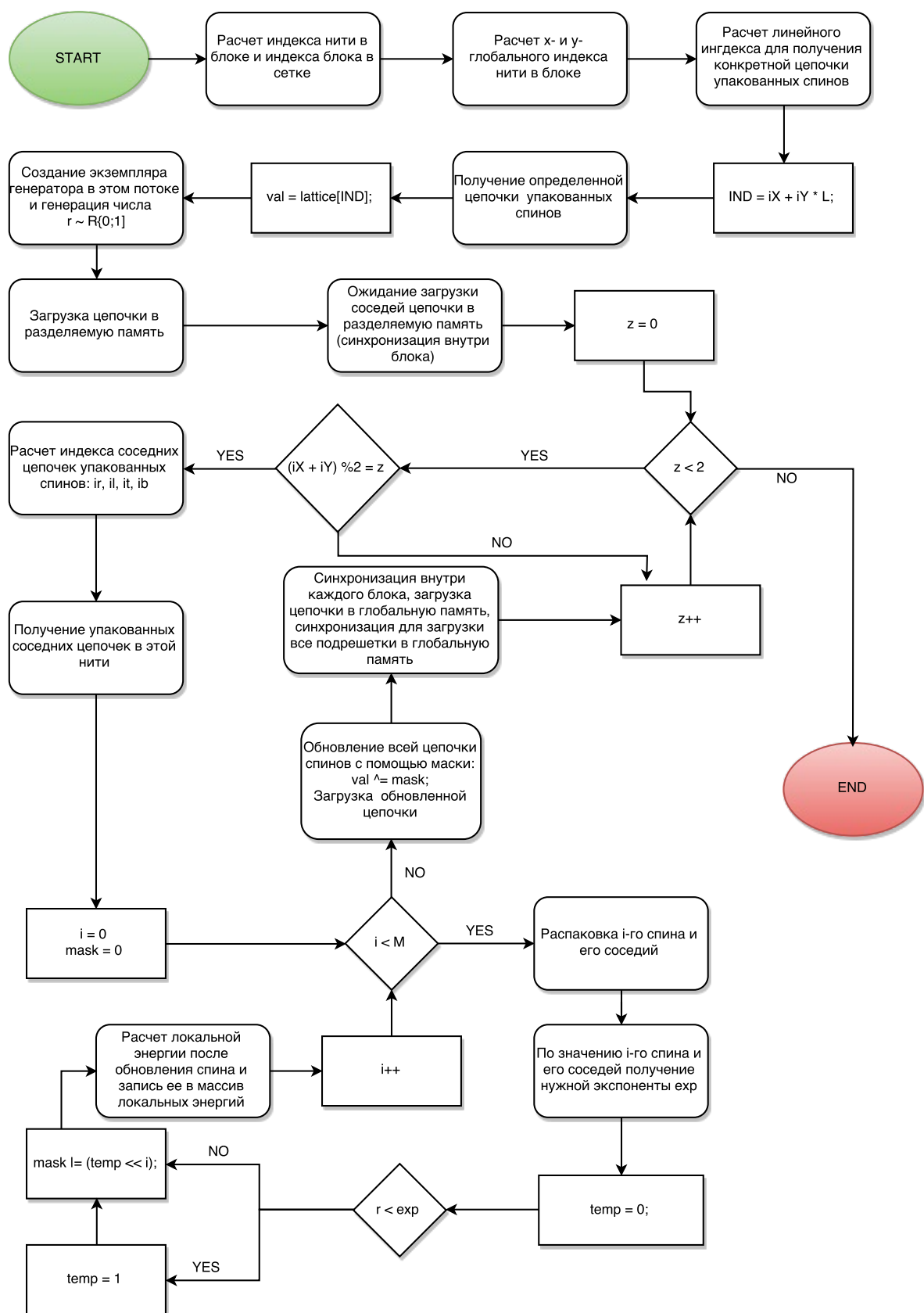


Рис. 5: Схема ядра для обновления спинов

4 Результаты моделирования

Данная часть работы посвящена анализу результатов моделирования. Как уже было сказано, в результате моделирования были получены следующие зависимости для двумерной модели Изинга (8) от температуры: внутренняя энергия, теплоемкость, намагниченность и магнитная восприимчивость. Моделирования происходило с использованием технологии упаковки AMSC и без него.

На графиках 6 - 9 представлены зависимости намагниченности, магнитной восприимчивости, внутренней энергии и теплоемкости от температуры для разных размеров решеток с AMSC и без него.

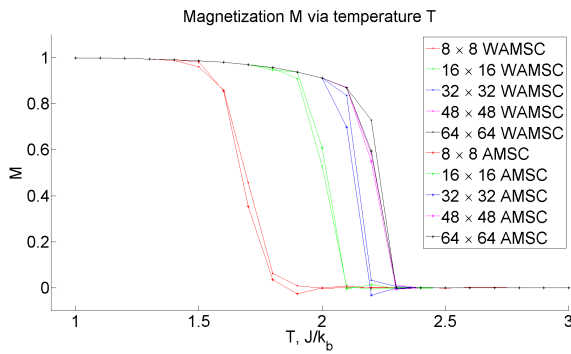


Рис. 6: Зависимость намагниченности от температуры при количестве шагов Монте-Карло 10^5

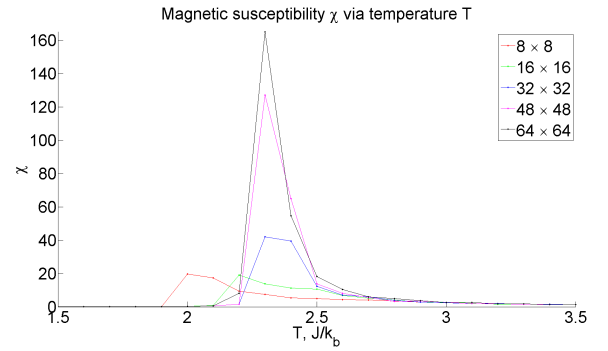


Рис. 7: Зависимость магнитной восприимчивости от температуры при количестве шагов Монте-Карло 10^5

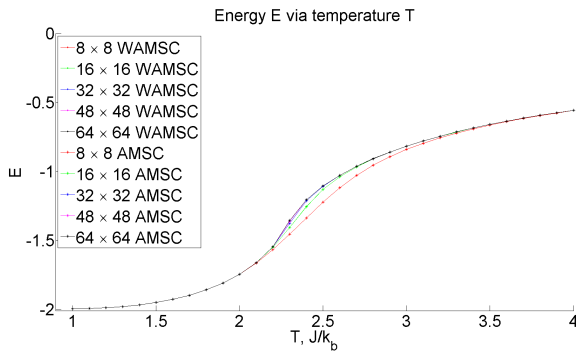


Рис. 8: Зависимость внутренней энергии от температуры при количестве шагов Монте-Карло 10^5

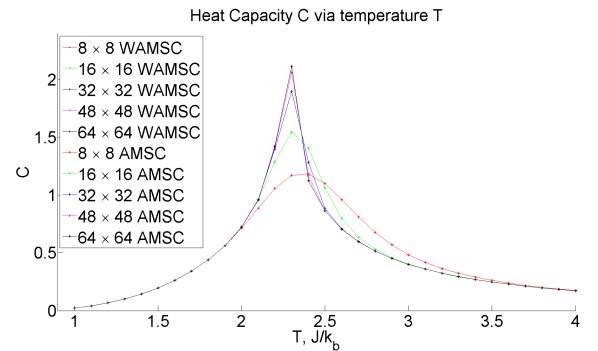


Рис. 9: Зависимость теплоемкости от температуры при количестве шагов Монте-Карло 10^5

Данные графики говорят о том, что модель Изинга - простейшая модель ферромагнетика. На графике 6 видно, что при температурах выше температуры Кюри

система имеет магнитные свойства - намагниченность близка к единице, вблизи такой температуры (критической области) модель резко теряет свои магнитные свойства, и при высоких температурах магнитные свойства полностью утрачены - намагниченность близка к нулю. Так же при увеличении размера решетки критическая область приближается к критической температуре. То есть для меньших размеров решетки система подвергается фазовому переходу быстрее по сравнению с более крупными размерами решетки.

Из графика 7 видно, что магнитная восприимчивость максимальна в критической области. Так же при увеличении размера решетки магнитная восприимчивость тоже увеличивается, при стремлении размера решетки к бесконечности, пик магнитной восприимчивости стремится тоже к бесконечности.

График 8 показывает, что при низких температурах модель находится в положении минимума внутренней энергии, все спины упорядочены. При температурах близких к критической температуре спины очень быстро теряют порядок, энергия возрастает. При высоких температурах энергия принимает максимально значение, спины полностью теряют порядок. Размер решетки влияет на скорость изменения внутренней энергии в критической области. С увеличением размера решетки производная функции внутренней энергии в критической области растет, так в пределе производная функции в критической области стремится к бесконечности.

График 9 говорит, что теплоемкость, как флуктуация энергии, максимальна вблизи критической точки. Так же из графика видно, что пик теплоемкости растет с размером решетки. Так же в критической области графики становятся более пологими с уменьшением размера решетки, т. е. с увеличением размера решетки теплоемкость меняется быстрее в критической области, то есть ее производная с увеличением размера решетки стремится к бесконечности в критической области.

Также необходимо показать, что при использовании всех оптимизаций время моделирования уменьшилось. Здесь представлено время работы одного запуска программы с использованием всех оптимизаций, без использования AMSC и CPU версии программы. В таблице 1 представлено время работы одного запуска программы при $MCS = 10^5$, а также во сколько раз быстрее происходило моделирование при использовании всех оптимизаций.

На основе данных из таблицы 1 можно сделать вывод, что моделирование с использованием всех оптимизаций дает большое ускорение по сравнению с обычной

Таблица 1: Время работы одного опыта при $MCS = 10^5$

| L | AMSC, t, сек | wAMSC, t, сек | CPU версия, t, сек | Ускорение, раз |
|----|--------------|---------------|--------------------|----------------|
| 8 | 14 | 21 | 33 | 2.357 |
| 16 | 29 | 40 | 129 | 4.448 |
| 32 | 65 | 97 | 519 | 7.985 |
| 48 | 86 | 130 | 1164 | 13.535 |
| 64 | 103 | 185 | 2134 | 20.718 |

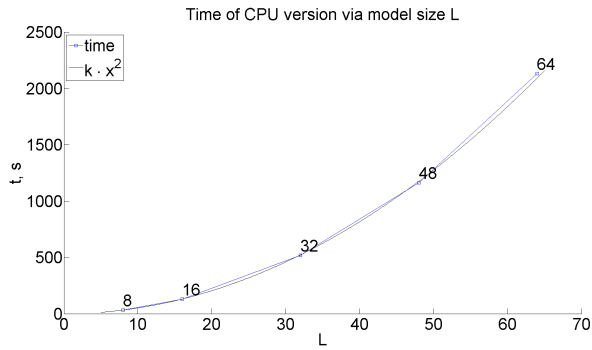


Рис. 10: Зависимость времени работы CPU программы от размера модели

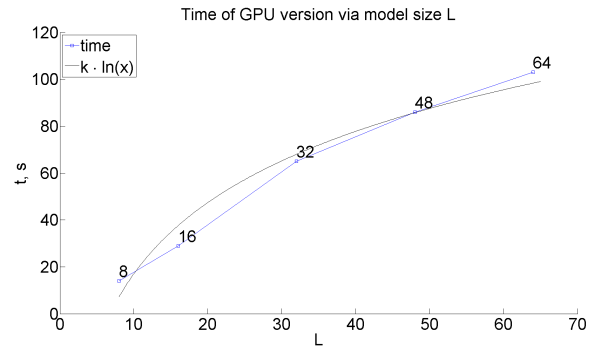


Рис. 11: Зависимость времени работы GPU программы от размера модели

версией программы. Также можно увидеть, что с увеличением размера решетки ускорение растет тоже. При больших размерах решетки, моделирования без использования GPU систем затруднительно, так как время одного опыта велико, а на практике необходимо около 200 таких запусков.

На графике 10 представлена зависимость времени одного запуска CPU версии программы от размера модели. Как можно видеть время работы растет квадратично с увеличением размера задачи. На графике 11 представлена также зависимость, только для GPU версии программы со всеми оптимизациями. Здесь время уже растет логарифмически, то есть размер задачи с какого то фиксированного размера перестает быть ключевым по время затратности.

Ускорение программы растет линейно с увеличением количества спинов L^2 в модели. Это видно из рисунка 12.

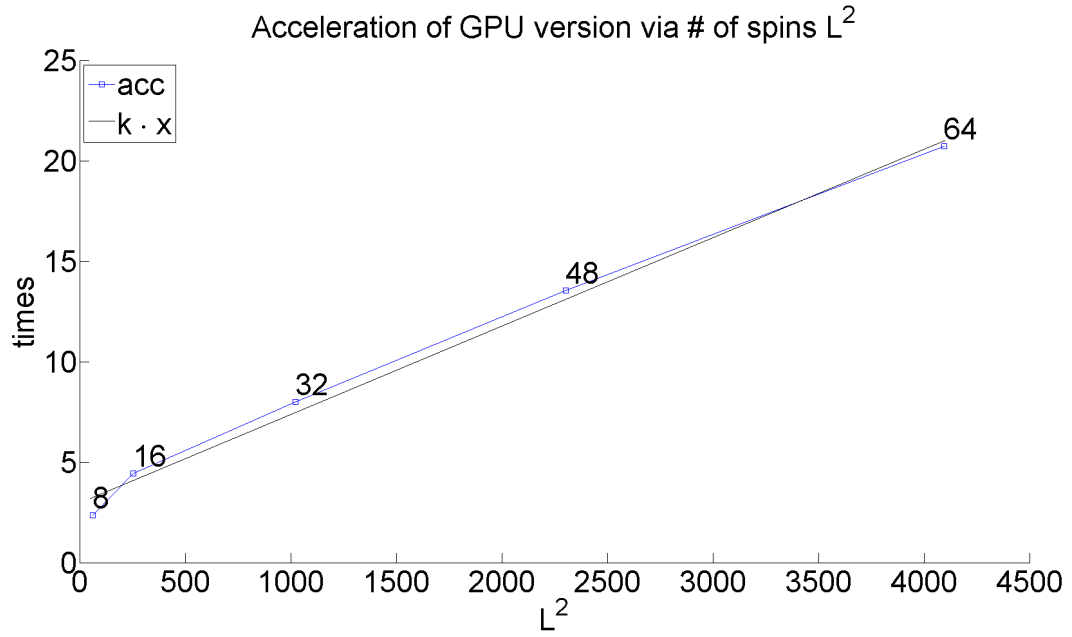


Рис. 12: Зависимость ускорения из таблицы 1 от количества спинов в модели L^2

4.1 Анализ результатов

Для проверки результатов моделирования я произвел сравнение их с аналитическими решениями, которые известны для модели Изинга. Аналитическое решение для модели Изинга конкретного размера изложено в работе [7]. В прикладном пакете *Wolfram Mathematica* были построены точные решения для внутренней энергии и теплоемкости для конкретного размера решетки. Отклонение внутренней энергии, полученной в результате моделирования, от точного значения вычислялась по формуле (22). Отклонение теплоемкости, полученной в результате моделирования, от точного значения вычислялась по формуле (23).

$$E_o = \frac{E_{exp} - E_{exact}}{E_{exact}} \quad (22)$$

$$C_o = \frac{C_{exp} - C_{exact}}{C_{exact}} \quad (23)$$

В формулах (22) и (23) величины с индексом *exp* - получены с помощью моделирования, *exact* - точные величины для конкретной температуры, построенные в прикладном пакете.

На графиках 13 - 22 показаны отклонения внутренней энергии от точного результата для разных размеров решетки с использованием AMSC и без него для разного количества шагов Монте-Карло.

На рисунках 23 - 32 изображено отклонение теплоемкости от точного результата для разных размеров решетки с использованием AMSC и без него для разного количества шагов Монте-Карло.

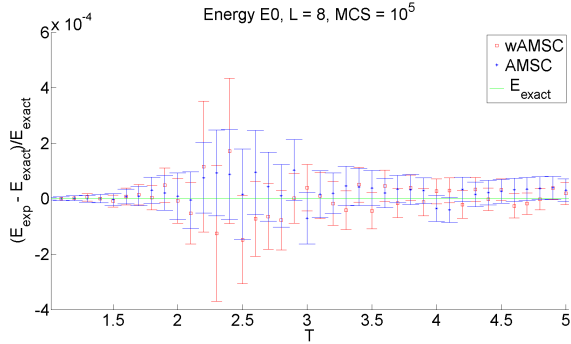


Рис. 13: Отклонение внутренней энергии для решетки 8×8 при количестве шагов Монте-Карло 10^5

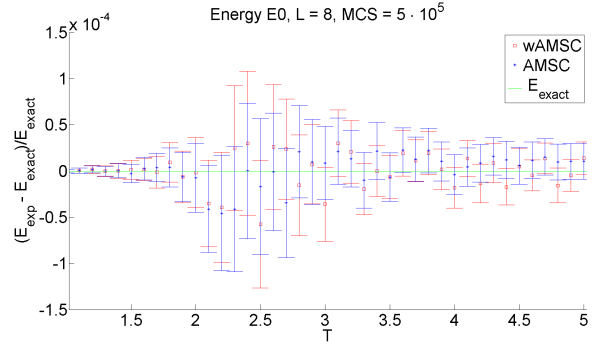


Рис. 14: Отклонение внутренней энергии для решетки 8×8 при количестве шагов Монте-Карло $5 \cdot 10^5$

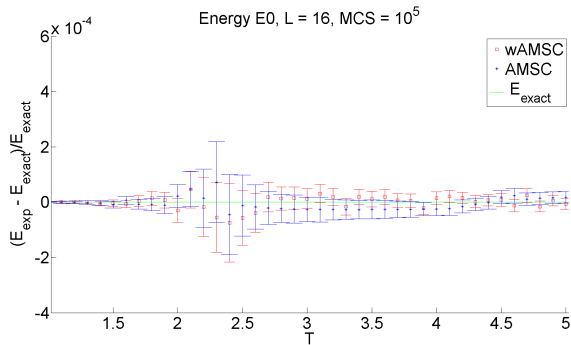


Рис. 15: Отклонение внутренней энергии для решетки 16×16 при количестве шагов Монте-Карло 10^5

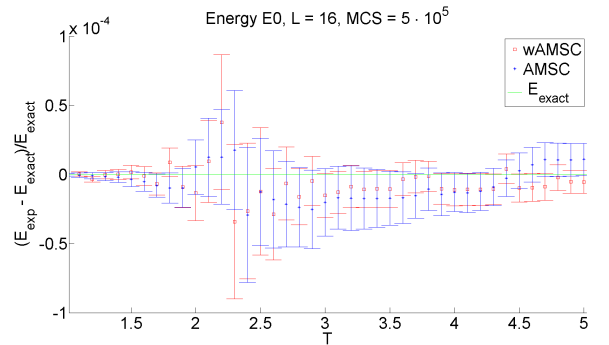


Рис. 16: Отклонение внутренней энергии для решетки 16×16 при количестве шагов Монте-Карло $5 \cdot 10^5$

На этих графиках изображен точный результат, полученный в прикладном пакете в виде горизонтальной линии, а результаты эксперимента в виде вертикальных линий - среднего значения внутренней энергии или теплоемкости в зависимости от графика при конкретной температуре и стандартное отклонение в виде планки погрешности. Из графиков видно, что с точностью до статистической погрешности внутренняя энергия и теплоемкость, полученные в результате моделирования, совпадают с точным значением для всех размеров решетки. Так же показано, что оптимизация AMSC не влияет на результаты моделирования.

На приведенных графиках также показана зависимость результатов моделирования от количества шагов Монте-Карло. При их увеличении результаты экс-

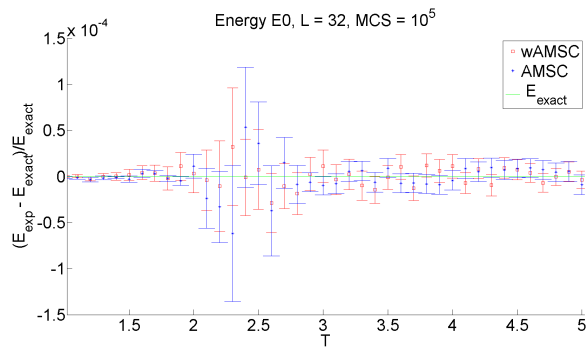


Рис. 17: Отклонение внутренней энергии для решетки 32×32 при количестве шагов Монте-Карло 10^5

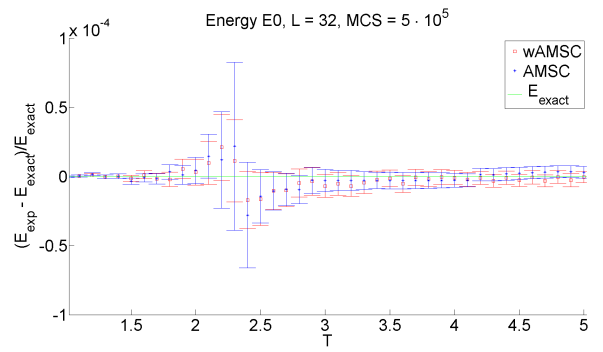


Рис. 18: Отклонение внутренней энергии для решетки 32×32 при количестве шагов Монте-Карло $5 \cdot 10^5$

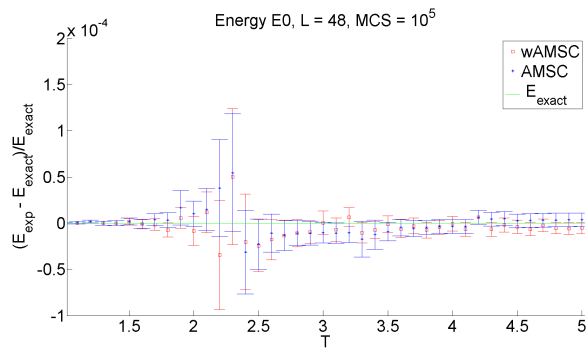


Рис. 19: Отклонение внутренней энергии для решетки 48×48 при количестве шагов Монте-Карло 10^5

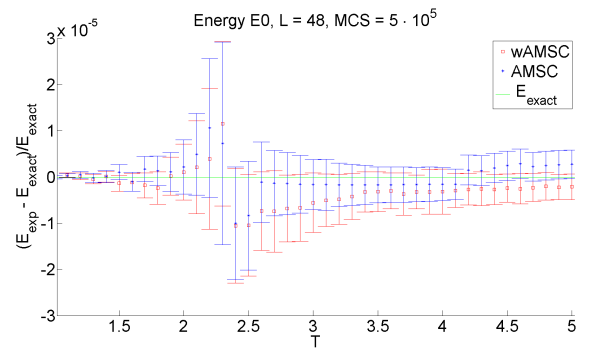


Рис. 20: Отклонение внутренней энергии для решетки 48×48 при количестве шагов Монте-Карло $5 \cdot 10^5$

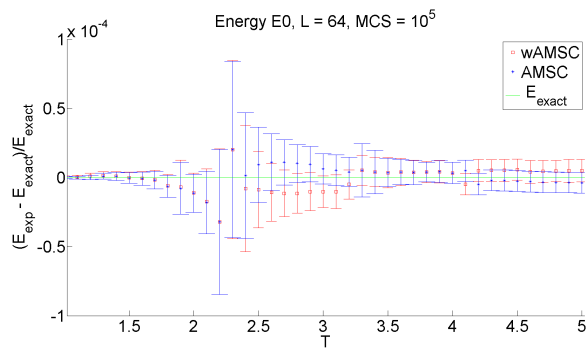


Рис. 21: Отклонение внутренней энергии для решетки 64×64 при количестве шагов Монте-Карло 10^5

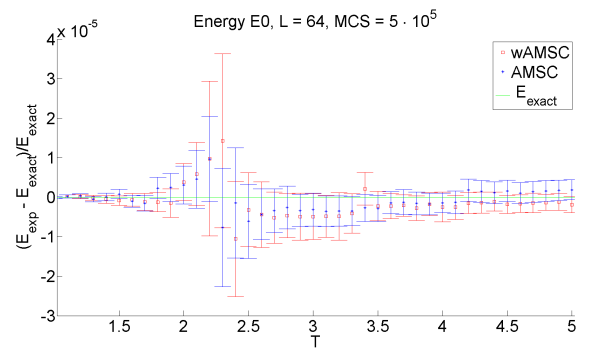


Рис. 22: Отклонение внутренней энергии для решетки 64×64 при количестве шагов Монте-Карло $5 \cdot 10^5$

перимента более точно совпадают с аналитическим результатом и статистическая ошибка тоже уменьшается.

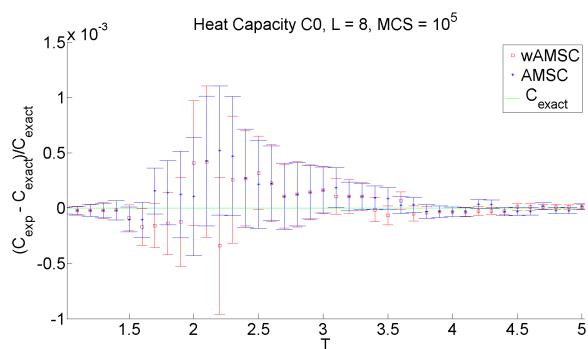


Рис. 23: Отклонение теплоемкости для решетки 8×8 при количестве шагов Монте-Карло 10^5

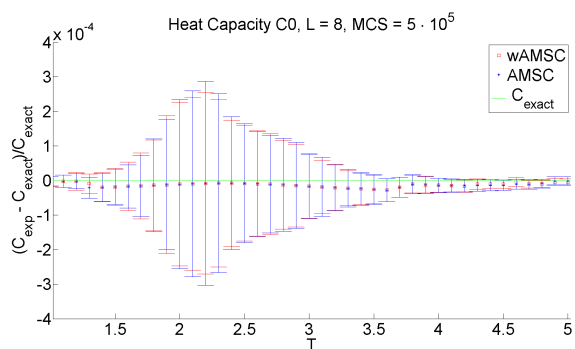


Рис. 24: Отклонение теплоемкости для решетки 8×8 при количестве шагов Монте-Карло $5 \cdot 10^5$

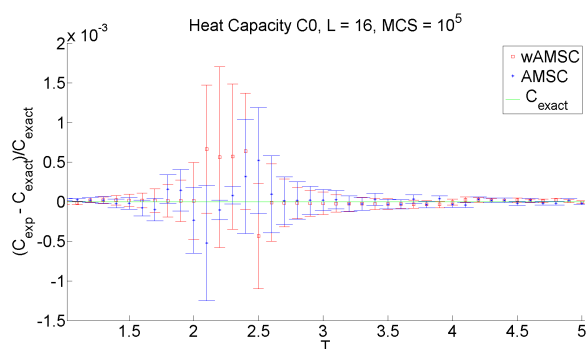


Рис. 25: Отклонение теплоемкости для решетки 16×16 при количестве шагов Монте-Карло 10^5

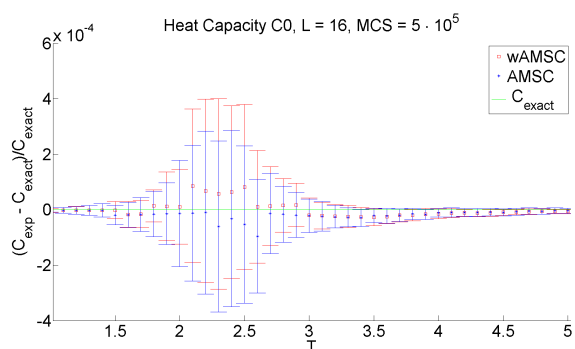


Рис. 26: Отклонение теплоемкости для решетки 16×16 при количестве шагов Монте-Карло $5 \cdot 10^5$

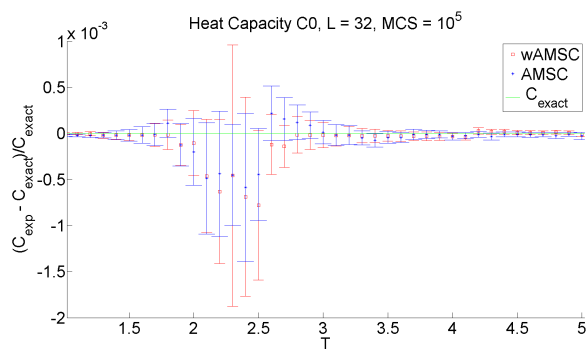


Рис. 27: Отклонение теплоемкости для решетки 32×32 при количестве шагов Монте-Карло 10^5

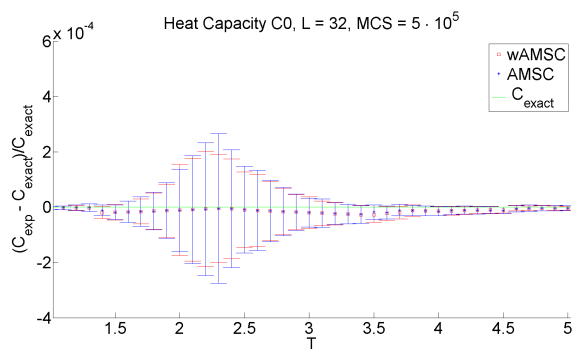


Рис. 28: Отклонение теплоемкости для решетки 32×32 при количестве шагов Монте-Карло $5 \cdot 10^5$

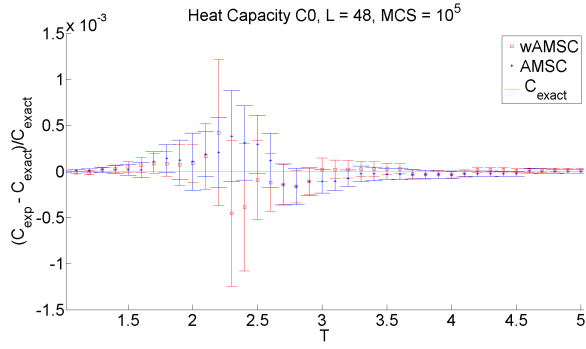


Рис. 29: Отклонение теплоемкости для решетки 48×48 при количестве шагов Монте-Карло 10^5

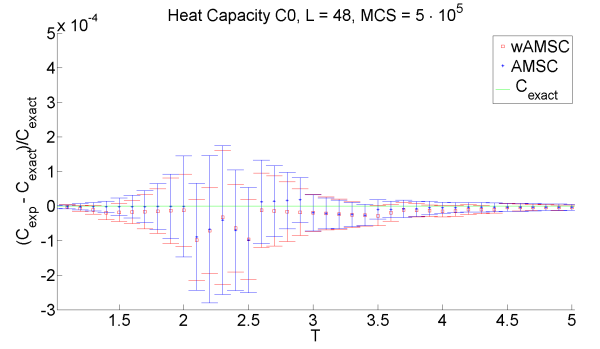


Рис. 30: Отклонение теплоемкости для решетки 48×48 при количестве шагов Монте-Карло $5 \cdot 10^5$

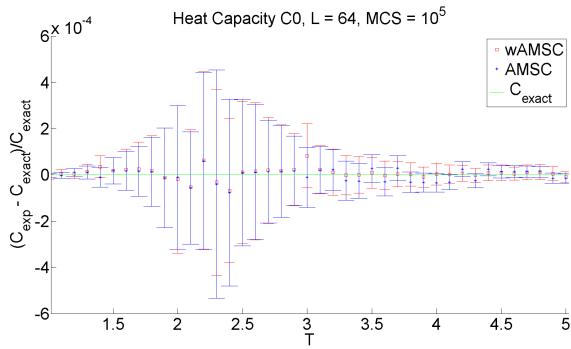


Рис. 31: Отклонение теплоемкости для решетки 64×64 при количестве шагов Монте-Карло 10^5

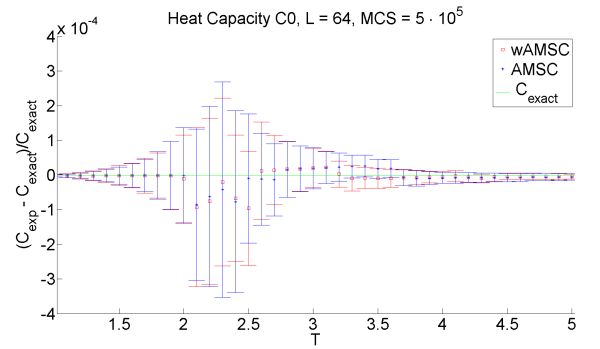


Рис. 32: Отклонение теплоемкости для решетки 64×64 при количестве шагов Монте-Карло $5 \cdot 10^5$

4.2 Эффекты размера модели

Существуют задачи, в которых влияние размеров модели на результаты моделирования существенно. Тогда для моделирования на компьютере необходимо определить такие размеры модели, которые будут вести себя в некоторых случаях как модели бесконечного размера [9].

В этой работе была проанализирована магнитная восприимчивость модели Изинга разных размеров. Для магнитной восприимчивости модели Изинга на решетке бесконечного размера есть точное решение, полученное в работе [8]. Магнитная восприимчивость для решетки бесконечного размера вычисляется:

$$T\chi = C_{0+} \left| 1 - \frac{T_c}{T} \right|^{-7/4} + C_{1+} \left| 1 - \frac{T_c}{T} \right|^{-3/4} \quad (24)$$

Где T - температура, χ - магнитная восприимчивость, $T_c = 2.26918531421$ -

температура фазового перехода. Уравнение (21) описывает магнитную восприимчивость при температуре выше температуры фазового перехода T_c . Коэффициенты следующие:

$$C_{0+} = 0.9625817322$$

$$C_{1+} = 0.0749881538$$

При увеличении размера решетки, магнитная восприимчивость, полученная в результате эксперимента, должна приближаться к магнитной восприимчивости решетки бесконечного размера. Введем параметр τ :

$$\tau = \frac{T - T_c}{T}$$

Тогда уравнение (24) примет вид:

$$\chi(\tau) = \left(C_{0+} |\tau|^{-7/4} + C_{1+} |\tau|^{-3/4} \right) \left(\frac{-T_c}{\tau - 1} \right)^{-1} \quad (25)$$

При некотором размере решетки, магнитная восприимчивость модели конечного размера должна практически совпадать с магнитной восприимчивостью модели бесконечного размера. Для этого происходило фитирование результатов эксперимента функцией определенного вида, и в зависимости от коэффициентов фитирования можно делать тот или иной вывод.

Так результаты эксперимента для магнитной восприимчивости можно фитировать функцией вида (26) и смотреть какие будут коэффициенты, так при определенном большом размере решетки, модель должна себя вести как модель бесконечного размера.

$$\chi(\tau)_L = \left(A |\tau|^{-7/4} + B |\tau|^{-3/4} \right) \left(\frac{-T_c}{\tau - 1} \right)^{-1} \quad (26)$$

Фитирование происходило только части полученных результатов в пределах $\tau_l < \tau < \tau_r$ [9], то есть той области, где эффекты размера минимальны. Значения пределов представлено в таблице 2.

С увеличением размера решетки область фитирования тоже увеличивается. Коэффициенты фитирования представлены в таблице 3.

С увеличением размера решетки коэффициенты фитирования приближаются к коэффициентам для решетки бесконечного размера. На рисунке 33 показано

Таблица 2: Значения τ_l и τ_r для разных размеров решетки

| L | τ_l | τ_r |
|----------|----------|----------|
| 8 | 0.1250 | 0.2 |
| 16 | 0.0625 | 0.2 |
| 32 | 0.0313 | 0.2 |
| 48 | 0.0208 | 0.2 |
| 64 | 0.0156 | 0.2 |
| 96 | 0.0104 | 0.2 |
| 128 | 0.0078 | 0.2 |
| 256 | 0.0039 | 0.2 |

| L | $C_{0_{exact}}$ | A | $C_{1_{exact}}$ | B |
|-----|-----------------|------------|-----------------|------------|
| 8 | 0.96258 | 0.18341(2) | 0.074988 | 2.1084(2) |
| 16 | 0.96258 | 0.26229(1) | 0.074988 | 1.8438(1) |
| 32 | 0.96258 | 0.41193(1) | 0.074988 | 1.0558(2) |
| 48 | 0.96258 | 0.48412(3) | 0.074988 | 0.43847(1) |
| 64 | 0.96258 | 0.51665(2) | 0.074988 | 0.31236(1) |
| 96 | 0.96258 | 0.72291(3) | 0.074988 | 0.20346(2) |
| 128 | 0.96258 | 0.82079(1) | 0.074988 | 0.1593(2) |
| 256 | 0.96258 | 0.96179(1) | 0.074988 | 0.07423(1) |

Таблица 3: Значение коэффициентов фитирования для разных размеров решетки

приближение результатов эксперимента для магнитной восприимчивости разных размеров решетки к решетке бесконечного размера. График построен в log-log масштабе. Как видно на графике 33 и в таблице 3, решетка размером 256×256 достаточно хорошо описывает решетку бесконечного размера.

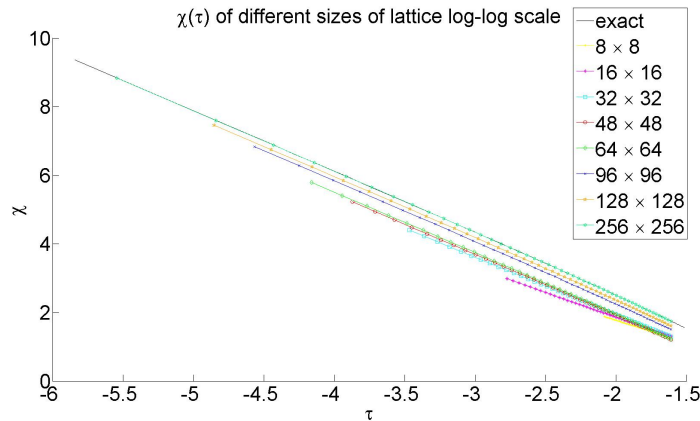


Рис. 33: Зависимость магнитной восприимчивости от параметра τ для разных размеров решетки

5 Выводы

В заключении проведенного нами исследования можно сделать следующие основные выводы по теме. Модель Изинга - одна из простых моделей статистической физики, описывающая поведение ферромагнетика. Данная модель может служить для проверки основных алгоритмов моделирования, так как в некоторых случаях для нее известно аналитическое решение.

Алгоритм декомпозиции типа двойной шахматной доски, основанный на алгоритме метрополиса, приведенный в этой работе, может быть применим и к другим спиновым моделям статистической физики с незначительными модернизациями.

Целью данного исследования была изучение двумерной модели Изинга и получение ее основных термодинамических характеристик. Для достижения этой цели были решены следующие задачи: была рассмотрена двумерная модель Изинга на квадратной решетки с периодическими граничными условиями и нулевым внешним магнитным полем. Был реализован оптимизированный алгоритм метрополиса для гибридного кластера и произведено моделирование. Были получены зависимости намагниченности, магнитной восприимчивости, внутренней энергии и теплоемкости от температуры для разных размеров модели и с разными параметрами моделирования. Также было произведено сравнение полученных результатов с аналитическим решением двумерной модели Изинга. Было показано, что результаты моделирования с точностью до статистической ошибки совпадают с точным решением для теплоемкости и внутренней энергии в двумерной модели Изинга. Так же были проанализированы эффекты размера и показано, что при некотором конечном размере модель может вести себя как модель бесконечного размера.

Это исследования может быть использовано для изучения фазового перехода в ферромагнетиках. Результаты этой работы согласуются с аналитическим решением модели и могут служить для понимания природы ферромагнетизма.

Благодарность

Хочу выразить благодарность Щуру Льву Николаевичу и Барашу Льву Юрьевичу за постановку и обсуждение задачи, помощь в реализации алгоритмов и полезные комментарии в ходе работы.

Список литературы

- [1] *Martin Weigel*, Simulating spin models on GPU, International Journal of Modern Physics C, Vol. 23, No. 8 (2012) 1240002 (11 pages)
- [2] *Ernst Ising*, Contribution to the Theory of Ferromagnetism, Zeitschrift für Physik, vol. XXXI (1925)
- [3] *Lars Onsager*, A Two-Dimensional Model with an Order-Disorder Transition, Phys. Rev. 65 (1944), 117-149
- [4] *D. P. Landau*, Finite-size behavior of the Ising square lattice, Phys. Rev. B 13, 2997 (1976), 138-153
- [5] *N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller*, Equation of state calculations by fast computing machines, J. Chem. Phys. 21 (1953), 1087-1092
- [6] *M. Lulli, M. Bernaschi, G. Parisi*, Highly optimized simulations on single- and multi-GPU systems of the 3D Ising spin glass model, Computer Physics Communications, 196(2015), 290-303
- [7] *Paul D. Beale*, Exact Distribution of Energies in the Two-Dimensional Ising Model, Physical Review Letters, Volume 76 (1996), 78-81
- [8] *T.T. Wu, B.M. McCoy, C.A. Tracy, and E. Barouch*, Spin-spin correlation functions for the two-dimensional Ising model: Exact theory in the scaling region, Phys. Rev. B 13 (1976), 316-374
- [9] *Lev N. Shchur, Oleg A. Vasilyev*, Critical amplitude ratio of the susceptibility in the random-site two-dimensional Ising model, PHYSICAL REVIEW E, VOLUME 65 (2001), 016107 (6 pages)
- [10] *A. В. Боресков, А. А. Харламов*, Основы работы с технологией CUDA , Издательство ДМК - Пресс, Москва 2011
- [11] *А. В. Боресков, А. А. Харламов, Н. Д. Марковский*, Параллельные вычисления на GPU, Суперкомпьютерный консорциум университетов России, Москва 2012
- [12] *M. Baity-Jesi, R. Baos, A. Cruz, L. Fernandez, J. Gil-Narvion, A. Gordillo-Guerrero, M. Guidetti, D. Iiguez, A. Maiorano, F. Mantovani, E. Marinari, V. Martin-Mayor, J. Monforte-Garcia, A. Muoz Sudupe, D. Navarro, G. Parisi, M. Pivanti, S. Perez- Gavio, F. Ricci-Tersenghi, J. Ruiz-Lorenzo, S. Schifano, B. Seoane, A. Tarancon, P. Tellez, R. Tripiccion, D. Yllanes* Reconfigurable computing for Monte Carlo simulations: Results and prospects of the Janus project, Eur. Phys. J. Spec. Top. 210 (1) (2012), 33–51
- [13] *Y. Fang, S. Feng, K.-M. Tam, Z. Yun, J. Moreno, J. Ramanujam, M. Jarrell*, Parallel tempering simulation of the three-dimensional Edwards–Anderson model with compact asynchronous multispin coding on GPU, Comput. Phys. Comm. (0) (2014), 2467–2478
- [14] *J-S Wang, W Selke, VB Andreichenko, and VS Dotsenko*, The critical behaviour of the two-dimensional dilute model, Physica A 164 (1990), 221–239

- [15] *L. Yu. Barash, L. N. Shchur*, PRAND: GPU accelerated parallel random number generation library: Using most reliable algorithms and applying parallelism of modern GPUs and CPUs, Computer Physics Communications, Volume 185, Issue 4 (2014), 1343–1353