

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра дискретной математики и информационных технологий

**РАЗРАБОТКА ЭЛЕМЕНТОВ СИСТЕМЫ ПОСТРОЕНИЯ
АДАПТИВНЫХ СЕТОК ДЛЯ ЗАДАЧ МОДЕЛИРОВАНИЯ
ПОВЕДЕНИЯ КВАНТОВЫХ СИСТЕМ ВО ВНЕШНИХ
ПОЛЯХ**

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

студента 2 курса 271 группы
направления 09.04.01 — Информатика и вычислительная техника
факультета КНиИТ
Новикова Николая Андреевича

Научный руководитель

к. ф.-м. н.

А. Д. Панферов

Заведующий кафедрой

к. ф.-м. н.

Л. Б. Тяпаев

Саратов 2021

ВВЕДЕНИЕ

Построение математических моделей является эффективным методом изучения различных явлений природы. Достаточно часто зависимость между различными характеристиками рассматриваемого физического, биологического или какого-нибудь другого динамического процесса не имеет прямого аналитического выражения, а определяется только в форме, например, дифференциальных уравнений, содержащих как текущие значения характеристик, так и их производных. Развитие вычислительной техники и численных алгоритмов решения уравнений различного вида сделало процедуру использования таких моделей достаточно обычной и успешной практикой.

Одним из приоритетных направлений современных прикладных исследований является поиск новых материалов и технологий их применения для достижения более высоких параметров электронных компонентов и устройств. Определенные надежды в этой связи возлагаются на псевдо двумерные структуры, в качестве наиболее известного представителя которых можно отметить графен (монослой углерода).

Основной целью выпускной квалификационной работы является развитие системы моделирования поведения графена во внешних электрических полях. Этот материал является квантовой системой и используемая математическая модель представляет собой версию квантового кинетического уравнения.

Выпускная квалификационная работа состоит из введения, трех глав, заключения и списка использованных источников.

Главы ВКР:

- Формулировка задачи.
- Инструменты для программной реализации.
- Реализация элементов системы моделирования.

Общий объем работы – 50 страниц, из них 34 страницы – основное содержание, включая 22 рисунка, список использованных источников содержит 30 наименований.

1 Основное содержание работы

Основной задачей стало совершенствование и развитие программного комплекса для моделирования процессов в графене в условиях действия внешнего электрического поля. Графен – одна из форм существования обычного углерода [1]. Фактически это гигантская поли-молекула, которая образуется атомами только одного элемента. Они образуют периодическую решетку с правильными шестигранными ячейками. Имея толщину всего в один слой атомов, эта решетка может иметь неограниченные размеры в двух других измерениях. Хотя в используемой для описания этого материала модели и фигурируют электроны, эти электроны очень непохожи на обычные электроны, известные физикам еще с конца девятнадцатого века. В этом материале свободные электроны, способные перемещаться под действием внешнего электрического поля, ведут себя как безмассовые частицы. В этом смысле они являются квазичастицами, т.е. не настоящими электронами, а результатом сложного квантовомеханического взаимодействия с окружающими их атомами углерода. Впервые эти свойства для графена были предсказаны в работе [2].

Для построения модели процессов, происходящих в графене под действием внешнего электрического поля, используется квантовая кинетическая теория [3]. Такой подход обеспечивает получение уравнения, описывающего процесс возникновения свободных носителей заряда и их эволюции, перехода множества частиц из одного состояния в другое. КУ представимо в форме системы обыкновенных дифференциальных уравнений (ОДУ) (1).

$$\begin{aligned} \dot{f}(\vec{p}, t) &= \frac{1}{2}\lambda(\vec{p}, t)u(\vec{p}, t), \\ \dot{u}(\vec{p}, t) &= \lambda(\vec{p}, t)(1 - 2f(\vec{p}, t)) - 2\varepsilon(\vec{p}, t)v(\vec{p}, t), \\ \dot{v}(\vec{p}, t) &= 2\varepsilon(\vec{p}, t)u(\vec{p}, t). \end{aligned} \tag{1}$$

Функция $f(\vec{p}, t)$ является функцией распределения и определяет заселенность состояний в импульсном пространстве. Через неё определяется пространственная плотность частиц и плотность тока проводимости:

$$\begin{aligned}
n(t) &= 4 \int \frac{d^2p}{(2\pi)^2} f(p_1, p_2, t), \\
j_1(t) &= 8e \int \frac{d^2p}{(2\pi)^2} \frac{p_1 - \frac{e}{c} A_1(t)}{\varepsilon(p_1, p_2, t)} f(p_1, p_2, t), \\
j_2(t) &= 8e \int \frac{d^2p}{(2\pi)^2} \frac{p_2 - \frac{e}{c} A_2(t)}{\varepsilon(p_1, p_2, t)} f(p_1, p_2, t).
\end{aligned} \tag{2}$$

Функции $u(\vec{p}, t), v(\vec{p}, t)$ носят вспомогательный характер и необходимы для вычисления поляризаационных эффектов.

Решение системы обыкновенных дифференциальных уравнений вида (1) – вычислительно сложная задача, и процесс получения множества точек, описывающих поведение функции распределения в некотором объёме импульсного пространства, требует больших вычислительных ресурсов. При этом необходимо обеспечить достаточно полное покрытие области импульсного пространства (p_1, p_2) .

Такое покрытие осуществляется с помощью выбора двумерной сетки. Сетка может быть равномерной, но такой выбор не оптимален. Программный комплекс, с которым велась работа, реализует многоэтапную процедуру построения адаптивной сетки, которая минимизирует количество необходимых узлов и, соответственно, требования к вычислительным ресурсам для решения задачи.

При решении задач оптимизации процедуры численного интегрирования часто применяется бисекция шага. Например, при использовании правила Рунге для оценки погрешности квадратурных формул и в адаптивных алгоритмах. Эта процедура обобщаема на двумерный и трехмерный случаи путем использования квадродеревьев. Для рассматриваемого двумерного пространства (p_1, p_2) квадродерево позволяет определить его полное покрытие квадратами, отношение сторон которых принадлежит ряду значений $2N$, где $N = 1, 2, 3, \dots$ ряд натуральных чисел. Значение функции в пределах каждого такого квадрата принимается равным её значению в центральной точке.

Построение адаптивной сетки выполняется путем поэтапного разбиения (квадрирования) выбранных её элементов. Элементы, не подвергавшиеся разбиению, являются листьями квадродерева. Элементы, над которыми процедура разбиения была выполнена, представляют из себя промежуточные

узлы. Критерии выбора элементов, подлежащих дальнейшему разбиению, будут играть ключевую роль для успешной работы алгоритма. Каждый шаг алгоритма будет переводить часть листьев последнего поколения в статус промежуточных узлов, надстраивая над ними листья нового поколения и завершаться вычислением значений функции распределения в новых листьях.

Для работы с квадродеревом был разработан программный комплекс, реализующий следующую последовательность этапов:

1. Шаг 1: "start_q_tree_gen.c" – стартовая генерация квадродерева. В файле "q_tree.txt" формируется квадродерево со стартовыми (в основном нулевыми) значениями полей.
2. Шаг 2: "calc_assign_points.c" – подсчет значений функции распределения в заданных узлах квадродерева. Вычисление значений функции распределения для каждой рассматриваемой точки импульсного пространства выполняется независимо в параллельном режиме с использованием ядер универсальных процессоров вычислительного кластера или ядер ускорителей NVIDIA. Результаты счета помещаются в файл "calc_resalt_temp.txt".

Стоит отметить, что строки подсчитанных значений в файле "calc_resalt_temp.txt" расположены хаотично. Такое нарушение порядка вывода результатов является классическим примером "гонки" независимых процессов и обычно рассматривается как ошибка организации параллельной секции программы. В данном случае отсутствие синхронизации вывода данных является осознанным решением, направленным на исключение дополнительных потерь времени. Упорядочение записей выполняется отдельным программным модулем.

3. Шаг 3: "calc_points_to_q_tree.c" – необходимая сортировка значений из файла "calc_resalt_temp.txt" с последующей интеграцией в файл "q_tree.txt".
4. Шаг 4: "calc_density.c" – подсчет плотности точек квадродерева. На этом шаге для контроля правильности получаемых результатов выполняется вычисление прогнозируемого значения плотности носителей по первой формуле формуле выражения (2).
5. Шаг 5: "add_new_points_to_q_tree.c" – Дополнение квадродерева новыми точками в файл "q_tree.txt".

Это итерационная процедура, которая заключается в последовательном дополнении квадродерева новыми точками и счетом значения функции распределения в них. Шаг 1 повторяется всего один раз, а шаги 2-5 повторяются циклически.

При проведении вычислительных экспериментов было обнаружено, что часто возникает ситуация, когда воспроизводимая функция распределения локализована в очень небольшой части импульсного пространства и построение квадродерева во всей первой зоне Бриллюэна $-\pi \leq p_1 \leq \pi, -\pi \leq p_2 \leq \pi$ неэффективно. Была сформулирована гипотеза, что для локализации области в импульсном пространстве, где необходимо точно воспроизводить поведение функции распределения, необходимо учитывать всю подобласть, где функция распределения за время действия электрического поля достигает значений, превышающих некоторый заданный порог $f_{max}(p_1, p_2, t) \geq f_{thr}$ (f_{thr} – заданное пороговое значение). Поэтому передо мной была поставлена задача разработать версию вычислительного модуля, который для каждой заданной точки импульсного пространства вычислял и выводил бы $f_{max}(p_1, p_2, t)$.

Во второй главе «Инструменты для программной реализации» производился обзор средств для реализации программного комплекса.

Были рассмотрены такие инструменты как:

1. Инструменты C/C++ для работы с файлами данных – необходимы для модификации программы интеграции результатов работы вычислительного модуля в основную файл квадродерева.
2. Алгоритмы сортировок – использовались для повышения эффективности разобы модуля интеграции.
3. Библиотека GSL – необходима для численного решения системы обыкновенных дифференциальных уравнений в вычислительном модуле.
4. Менеджер распределённых ресурсов TORQUE – необходим для запуска программного комплекса на кластере СГУ.
5. Message Passing Interface – используется для организации параллельной работы вычислительного модуля.
6. Инструменты визуализации – описывают средства GNU Plot, необходимые для визуализации результатов.

В заключительной главе «Реализация элементов системы моделирования» описывается процесс решения практических задач.

Первой практической задачей была реализация процедуры сортировки и записи выходных данных вычислительного модуля в основной файл квадродерева. Целью разработки такой процедуры была замена старой версии аналогичной программы по причине ее неэффективности. Для этого на языке C++ был написан программный модуль, который с помощью функций произвольного доступа к файлу читал файл "calc_result_temp.txt", сортировал хаотично записанные в него строки с необходимыми данными и записывал эти данные в соответствующие позиции основного файла квадродерева.

Для сравнения быстродействия был проведен ряд тестов старой и новой версии программы. Для этого были построены все доступные варианты стартового квадродерева: от одного до семи поколений. А количество точек, в которых вычислялись значения искомым функций варьировалось от 4 до 21844. За количество поколений и, соответственно, число точек отвечает соотношение параметров "Ширина рассматриваемой области" и "Масштаб сплошного покрытия области" в файле характеристик квадродерева "task_q.txt".

По результатам сравнения можно сделать вывод, что обновленная версия программы, использующая инструменты произвольного доступа к файлу и сортировку промежуточных значений, дает выигрыш в быстродействии обработки 21844 точек примерно в 200 раз. Результаты запуска программ для 7 поколений квадродерева можно назвать самыми точными, потому что разработанная мной программа требует всего 15 секунд для обработки 21844 точек. Старая версия программы обрабатывала 7 поколений квадродерева около 50 минут, что подчеркивает неэффективность используемого ранее алгоритма.

Следующим этапом была реализация пакетного запуска всего программного комплекса. Необходимость такой процедуры обусловлена удобством работы с квадродеревом. Предыдущая версия программного комплекса использовала последовательный запуск программ построения квадродерева, вычислительного модуля и модуля сортировки по отдельности. Пакетный запуск позволяет проводить всю процедуру работы с квадродеревом за один запуск скрипта. Для решения этой проблемы был написан bash-скрипт последовательного запуска задач с использованием функционала менеджера распределённых ресурсов TORQUE. Он позволяет отслеживать завершение предыдущей задачи, и только тогда запускать следующую. Таким образом, весь ком-

плекс программ может быть запущен одной командой.

Третьим элементом системы моделирования, который был мной существенно переработан был основно вычислительный модуль, реализующий решение системы уравнений в параллельном режиме. В исходной версии модуль обеспечивал вычисление конечных значений функции распределения $f_{end}(p_1, p_2)$ и вспомогательных функций $u_{end}(p_1, p_2)$ и $v_{end}(p_1, p_2)$ для заданного массива точек (p_1, p_2) . При разработке процедуры локализации области относительно больших значений функции распределения [28] использовалась модифицированная версия вычислительного модуля, вычислявшая значения искомой функции для двух моментов времени. Первый момент времени соответствовал максимальному значению напряженности внешнего поля, а второй – конечному состоянию системы. В результате анализа полученных данных было сделано предположение, что наиболее эффективный способ локализации области максимальных значений конечных значений функции распределения – определять ее по распределению максимальных значений функции распределения, которые она достигает за все время действия электрического поля. Передо мной была поставлена задача модифицировать вычислительный модуль, чтобы в ходе его работы рассматривались и анализировались все промежуточные значения функции распределения, определялось и выводилось максимальное из них. Для этого была реализована пошаговая процедура вычисления промежуточных значений функции распределения для задаваемого ряда эквидистантных точек на временной оси и их сравнения. Это реализовано средствами управления процедурой решения задачи Коши библиотеки SGL.

Для визуализации результатов использовались трехмерные графики и картирование, построенные с помощью средств GNU Plot.

ЗАКЛЮЧЕНИЕ

В ходе работы все поставленные задачи были решены. Был проведен обзор таких инструментов как произвольный доступ к файлам в языке C++, алгоритмы сортировок, менеджер распределенных ресурсов TORQUE и примеры его запуска на кластере СГУ. Также был рассмотрен MPI как средство обмена информацией между параллельными процессами и GNU Plot как средство визуализации полученных данных.

На практике была модифицирована программа объединения выходных данных вычислительного модуля с основным файлом квадродерева. В основе вычислительного модуля лежит численное решение системы обыкновенных дифференциальных уравнений средствами библиотеки GSL языка C. Эффективность новой версии программы превосходит старую в 200 раз. Кроме того, был модифицирован сам вычислительный модуль, что позволило визуализировать максимальные значения функции распределения средствами GNU Plot. Для визуализации результатов использовались трехмерные графики и двумерное картирование.

Кроме того, для подготовки данных к визуализации были разработаны две вспомогательные программы, которые извлекают необходимые данные из квадродерева и выводит их в необходимом формате для отрисовки в линейной и логарифмической шкале.

Заключительным этапом было объединение всего программного комплекса в один пакетный файл. Задача была успешно выполнена, после чего программный комплекс стало возможным запускать одним скриптом с использованием средств TORQUE.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Графен [Электронный ресурс] : Материал из Википедии – свободной энциклопедии : / Авторы Википедии // Википедия, свободная энциклопедия. – Электрон. дан. – Сан-Франциско: Фонд Викимедиа, 2019. – URL: <https://ru.wikipedia.org/?oldid=100220118> (дата обращения: 10.05.2019) – Версия 100220118, сохранённая в 10:44 UTC 4 июня 2019 – Загл. с экрана. Яз. рус.
- 2 P.R. Wallace, Phys. Rev. 71, p. 622-634 (1947)
- 3 Smolyansky S.A., Panferov A.D., Blaschke B.D., and Gevorgyan N.T. Nonperturbative Kinetic Description of Electron-Hole Excitations in Graphene in a Time Dependent Electric Field of Arbitrary Polarization // Particles 2019, 2, 208-230.
- 4 P. Bowlan, E. Martinez-Moreno, K. Reimann, T. Elsaesser, M. Woerner. “Ultrafast terahertz response of multilayer graphene in the nonperturbative regime”, Phys. Rev. B, 89 (2014), 041408.
- 5 D. H. Bailey, J. M. Borwein. “High-precision arithmetic in mathematical physics”, Mathematics, 3:2 (2015), pp. 337–367.
- 6 A. S. Korzhavina, V. S. Knyaz’kov. “High-precision computations using residueinterval arithmetic on FPGAs”, Program Systems: Theory and Applications, 10:3(42) (2019), pp. 81–127 (in Russian).
- 7 A. D. Panferov, A. V. Makhankov, A. V. Trunov. “The use of an adaptive mesh based on a quadtree for modeling the final state of a quantum field system under pulsed external action”, Program Systems: Theory and Applications, 11:1(44) (2020), pp. 93–105.
- 8 S. A. Smolyansky, D. V. Churochkin, V. V. Dmitriev, A. D. Panferov, B. Kampfer. “Residual currents generated from vacuum by an electric field pulse in 2+1 dimensional QED models”, EPJ Web of Conferences, 138 (2017), 06004, 5 pp.
- 9 D. B. Blaschke, B. Kampfer, S. M. Schmidt, A. D. Panferov, A. V. Prozorkevich, S. A. Smolyansky. “Properties of the electron-positron plasma created from a vacuum in a strong laser field: Quasiparticle excitations”, Phys. Rev. D, 88 (2013), 045017.

- 10 A. D. Panferov, S. A. Smolyansky, A. Otto, B. Kampfer, D. B. Blaschke, L. Juchnowski. “Assisted dynamical Schwinger effect: pair production in a pulsed bifrequent field”, *Eur. Phys. J. D*, 70 (2016), 56.
- 11 N. Yoshikawa, T. Tamaya, K. Tanaka. “High-harmonic generation in graphene enhanced by elliptically polarized light excitation”, *Science*, 356:6339 (2017), pp. 736–738.
- 12 M. Taucer, T. J. Hammond, P. B. Corkum, G. Vampa, C. Couture, N. Thire, B. E. Schmidt, F. Legare, H. Selvi, N. Unsuree, B. Hamilton, T. J. Echtermeyer, M. A. Denecke. “Nonperturbative harmonic generation in graphene from intense midinfrared pulsed light”, *Phys. Rev. B*, 96 (2017), 195420.
- 13 S. A. Smolyansky, A. D. Panferov, D. B. Blaschke, N. T. Gevorgyan. “Kinetic equation approach to graphene in strong external fields”, *Particles*, 3:2 (2020), pp. 456–476.
- 14 Поточковый ввод-вывод в файлы [Электронный ресурс]. – URL: <https://prog-cpp.ru/cpp-files/> (дата обращения: 1.11.2019). Загл. с экрана. Яз. рус.
- 15 Типы данных C++ [Электронный ресурс]. – URL: <http://cppstudio.com/post/271/> (дата обращения: 17.10.2019). Загл. с экрана. Яз. рус.
- 16 Ввод-вывод с использованием потоковых классов. Произвольный доступ к файлам [Электронный ресурс]. – URL: http://it.kgsu.ru/C_STL/c_stl496.html (дата обращения: 6.12.2019). Загл. с экрана. Яз. рус.
- 17 Сортировка [Электронный ресурс] : Материал из Википедии – свободной энциклопедии : Версия 104590621, сохранённая в 17:29 UTC 16 января 2020 / Авторы Википедии // Википедия, свободная энциклопедия. – Электрон. дан. – Сан-Франциско: Фонд Викимедиа, 2020. – Режим доступа: <https://ru.wikipedia.org/?oldid=104590621>
- 18 Кнут, Дональд, Эрвин. Искусство программирования, том 3. Сортировка и поиск, 2-е изд. : Пер. с англ. – М. : ООО "И.Д. Вильямс 2007. – 832 с.
- 19 Алгоритм сортировки [Электронный ресурс] : Материал из Википедии – свободной энциклопедии : Версия 106767196, сохранённая в 15:32 UTC 3

- мая 2020 / Авторы Википедии // Википедия, свободная энциклопедия. – Электрон. дан. – Сан-Франциско: Фонд Викимедиа, 2020. – Режим доступа: <https://ru.wikipedia.org/?oldid=106767196>
- 20 Хабр [Электронный ресурс] // <https://habr.com/ru/> – Сравнение алгоритмов сортировки URL: <https://habr.com/ru/post/274017/> (дата обращения: 07.06.2020) Загл. с экрана. Яз. англ.
- 21 GNU Scientific Library // Википедия. [2020]. Дата обновления: 21.07.2020. URL: <https://ru.wikipedia.org/?curid=1796112&oldid=108300153> (дата обращения: 21.07.2020).
- 22 Bauman National Library [Электронный ресурс] // <https://ru.bmstu.wiki> – GNU Scientific Library URL: https://ru.bmstu.wiki/GNU_Scientific_Library (дата обращения: 07.06.2020) Загл. с экрана. Яз. рус.
- 23 TORQUE (программа) // Википедия. [2019]. Дата обновления: 19.02.2019. URL: <https://ru.wikipedia.org/?curid=2952062&oldid=98217219> (дата обращения: 19.12.2020).
- 24 Portable Batch System // Википедия. [2019]. Дата обновления: 04.11.2019. URL: <https://ru.wikipedia.org/?curid=2969851&oldid=103134075> (дата обращения: 19.12.2020).
- 25 Torque: использование // Linux. Кластер. URL: <https://cluster.linux-ekb.info/torque2.php> (дата обращения: 19.12.2020).
- 26 Message Passing Interface // Википедия. [2020]. Дата обновления: 10.12.2020. URL: <https://ru.wikipedia.org/?curid=47119&oldid=110989931> (дата обращения: 10.12.2020).
- 27 Parallel [Электронный ресурс] // <https://parallel.ru/> – Message Passing Interface (MPI) URL: <https://parallel.ru/vvv/mpl.html> (дата обращения: 07.06.2020) Загл. с экрана. Яз. рус.
- 28 А.А. Трунов, Р.Д. Аль-Карави, Т.Т. Веревин, Н.А. Новиков, А.Д. Панферов. «Визуализация и анализ массивов данных при моделировании поведения графена во внешнем электрическом поле», Информационные технологии и математическое моделирование, Материалы XVIII Международной конференции имени А. Ф. Терпугова. Т. 1, ИТММ-2019 (26–30 июня 2019 г.,

Саратов, Россия), Издательство НТЛ, Томск, 2019, ISBN 978-5-89503-628-0, с. 158–162.

29 C++ [Электронный ресурс] // <https://proginfo.ru/> – Время URL: <https://proginfo.ru/time/> (дата обращения: 07.06.2020) Загл. с экрана. Яз. англ.

30 Gnuplot // Википедия. [2019]. Дата обновления: 12.06.2019. URL: <https://ru.wikipedia.org/?curid=32711&oldid=100379584> (дата обращения: 12.06.2021).