

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
имени М. В. ЛОМОНОСОВА

ФИЗИЧЕСКИЙ ФАКУЛЬТЕТ

Кафедра физики полимеров и кристаллов

Курсовая работа

Распараллеливание на GPU метода DPD  
для моделирования полимерных систем

Выполнил  
студент 2 курса 204 группы  
Байназаров Руслан Рысбекович

Научный руководитель  
Рудяк Владимир Юрьевич

---

19 мая 2016 г.

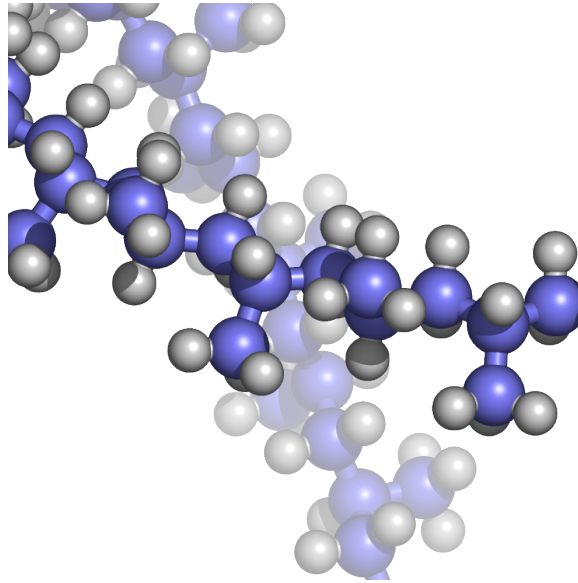
Москва, 2016 г.

## Содержание

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Введение</b>                                | <b>2</b>  |
| 1.1      | Методы моделирования полимеров . . . . .       | 3         |
| 1.1.1    | Метод Монте-Карло . . . . .                    | 3         |
| 1.1.2    | Молекулярная динамика . . . . .                | 4         |
| 1.1.3    | DPD . . . . .                                  | 5         |
| 1.2      | Метод интегрирования . . . . .                 | 7         |
| 1.3      | Распараллеливание DPD . . . . .                | 7         |
| 1.4      | Технические особенности работы с GPU . . . . . | 9         |
| <b>2</b> | <b>Результаты</b>                              | <b>10</b> |
| 2.1      | Моделирование известных систем . . . . .       | 10        |
| 2.1.1    | Расслоение мономеров . . . . .                 | 10        |
| 2.1.2    | Переход клубок – глобула . . . . .             | 13        |
| 2.2      | Полученное ускорение . . . . .                 | 15        |
| <b>3</b> | <b>Выводы</b>                                  | <b>17</b> |
|          | <b>Список литературы</b>                       | <b>17</b> |

## 1 Введение

Полимерная цепь – это цепь из большого числа мономерных элементов  $N$ , где  $N \gg 1$ . Полимерные молекулы или полимеры – это молекулы, состоящие из полимерных цепей. Вещества, состоящие из полимеров называют полимерными материалами. Полимерной цепью также называют математическую модель полимерной молекулы.



Некоторые свойства полимеров можно описать теоретически, на основе подходов статистической физики. К примеру, размер полимерной цепи зависит от ее длины как  $R = l^{1/2}L^{1/2}$ , где  $l$  – характерный параметр размерности длины. Эта аналитическая формула дает описание поведения полимерной цепи при ее формировании. Тем не менее, многие свойства не могут быть вычислены аналитически. Для полного описания полимерной системы необходимо решить статистическую задачу, что довольно сложно для таких систем, как разветвленные полимеры, гребнеобразные полимеры, полимерные сетки и др. Так же, не все свойства полимерных систем могут быть определены экспериментально: некоторые вещества могут не существовать или существовать в недостаточных для проведения эксперимента количествах. Внутреннюю структуру вещества обычно сложно изучить в эксперименте из-за малости масштабов.

Для предсказания большого количества свойств применяется так называемое «компьютерное моделирование». Этот термин применяется в случае, когда компьютер применяется не просто как средство для ускорения вычислений по готовым формулам, а как средство для имитации исследуемого процесса, в итоге которой извлекается информация об искомым свойствах системы. [2][4]

Компьютерное моделирование позволяет изучить такие свойства полимерных систем, которые не доступны для изучения другими способами. Фактически, проводится «численный экс-

перимент». Отличие такого эксперимента от реального в гораздо большей гибкости настроек начальных параметров системы и в возможности проведения практически любых измерений, в том числе и микроскопических. Еще одной важной функцией компьютерного моделирования является визуализация микроскопической структуры вещества. Это сильно облегчает понимание строения и свойств того или иного материала и их теоретическое описание. С помощью компьютерного моделирования можно предсказать свойства даже тех веществ, которые еще не были получены на практике. Численный эксперимент обычно намного дешевле и быстрее реального эксперимента, что позволяет получать большое количество информации из компьютерных расчетов.

Из всего вышесказанного можно сделать вывод, что компьютерное моделирование является важным инструментом исследования полимеров.

В настоящее время требуется моделировать очень большие системы ( $10^5$ ,  $10^6$  частиц) с большим временем релаксации. Примером таких систем могут послужить композитные материалы, состоящие из эластомеров и наполнителя[8]. Моделирование таких систем требует больших временных затрат и, поэтому, необходимо оптимизировать методы компьютерного моделирования.

Целью данной работы является адаптация метода диссипативной динамики частиц (DPD) для вычислений на графических картах с использованием технологии Nvidia CUDA для моделирования полимерных систем.

## 1.1 Методы моделирования полимеров

Существует множество методов моделирования полимерных систем, которые могут быть применены на разных пространственных и временных масштабах и с разной точностью.

### 1.1.1 Метод Монте-Карло

Идея метода Монте-Карло состоит в замене интеграла вида

$$\int_{\Omega} g(x_1, x_2, \dots, x_n) dx_1 dx_2 \dots dx_n$$

от некоторых переменных математическим ожиданием функции  $g(\xi)$  случайной величины  $\xi$ , равномерно распределенной в произвольной области  $\Omega$ . Принципиальную основу использования метода МК составляет теорема, называемая усиленным законом больших чисел в форме Колмогорова: для того, чтобы среднее арифметическое независимых реализаций случайной величины  $\xi$  сходилось к ее математическому ожиданию, необходимо и достаточно, чтобы это математическое ожидание существовало. Этот закон позволяет сделать и более общую замену.

Число

$$J = \int_{\Omega} g(x_1, x_2, \dots, x_n) \cdot f(x_1, x_2, \dots, x_n) dx_1 dx_2 \dots dx_n$$

заменяется его математическим ожиданием, которое сходится к среднему арифметическому

$$\langle J \rangle = \frac{1}{M} \sum_{k=1}^M g(x_1^k, x_2^k, \dots, x_n^k).$$

Этот принцип позволяет вычислять среднее значение некоторой физической наблюдаемой величины. Возьмем, к примеру молекулярный ансамбль  $N$  частиц с гамильтонианом взаимодействия  $U(\mathbf{x})$ , где  $\mathbf{x}$  – точка в фазовом пространстве  $\Omega$ . Среднее значение некоторой наблюдаемой величины  $H(\mathbf{x})$  можно вычислить как

$$\langle H \rangle = \frac{\int_{\Omega} H(\mathbf{x}) \cdot \exp\left(\frac{U(\mathbf{x})}{k_B T}\right) d\mathbf{x}}{\int_{\Omega} \exp\left(\frac{U(\mathbf{x})}{k_B T}\right) d\mathbf{x}}$$

Нужно подчеркнуть, что координаты и импульсы входят в гамильтониан по отдельности. Поэтому переменные разделяются, а интеграл по импульсной части берется аналитически тривиальным образом. Поэтому при обсуждении метода МК под фазовым пространством всегда подразумевается его конфигурационная часть.[4]

Таким образом, методом Монте-Карло можно вычислять параметры молекулярной системы путем вычисления их среднего значения по различным конфигурациям.

### 1.1.2 Молекулярная динамика

Идея молекулярной динамики состоит в сопоставлении реальному молекулярному объекту математического образа системы взаимодействующих материальных точек, движение которых описывается классическими уравнениями Ньютона:

$$m_i \frac{d^2 \mathbf{r}_i}{dt^2} = \mathbf{F}_i, \quad i = 1, \dots, N.$$

Обычно материальная точка представляет собой отдельный атом. Тогда  $N$  - это число атомов в моделируемой системе. Эти уравнения можно численно проинтегрировать и получить траектории движения всех частиц.

Между молекулами в реальных телах действуют различные силы. В компьютерном эксперименте взаимодействия аппроксимируются потенциальными функциями, зависящими от значения координат всех частиц.

$$\mathbf{F}_i = -\frac{\delta U(\mathbf{r}_1, \dots, \mathbf{r}_N)}{\delta \mathbf{r}_i}.$$

Потенциал  $U(\mathbf{r}_1, \dots, \mathbf{r}_N)$  выбирается из физических соображений. Существует множество различных силовых полей, наиболее распространенными являются AMBER, GROMOS, OPLS, MMFF, PCFF. Обычно, потенциал силового поля складывается из нескольких компонент:

$$U = U_{\text{валентных связей}} + U_{\text{валентных углов}} + U_{\text{торсионных углов}} + U_{\text{плоских групп}} + U_{VdW} + U_{qq}$$

Эти компоненты соответствуют различным видам взаимодействий и задаются различными формулами:

$$\begin{aligned} U_{\text{валентных связей}} &= \sum_i K_{l,i} (l_i - l_0)^2, \\ U_{\text{валентных углов}} &= \sum_i K_{\theta,i} (\theta_i - \theta_0)^2, \\ U_{\text{торсионных углов}} &= \sum_i K_{\phi,i} [1 \pm \cos(n_i \phi_i)], \\ U_{\text{плоских групп}} &= \sum_i K_{\psi,i} [1 - \cos(2\psi_i)]. \end{aligned}$$

Здесь  $l_i$ ,  $\theta_i$ ,  $\phi_i$ ,  $\psi_i$  – валентные связи, валентные углы, торсионные углы и углы неплоских отклонений соответственно;  $l_0$  и  $\theta_0$  – равновесные валентные связи и углы;  $K_{l,i}$ ,  $K_{\theta,i}$ ,  $K_{\phi,i}$ ,  $K_{\psi,i}$  – силовые постоянные;  $n_i$  – целые числа, отражающие симметрию потенциалов соответствующих торсионных углов.

Невалентные ван-дер-ваальсовские взаимодействия задаются в виде потенциала Леннарда-Джонса:

$$U_{VdW} = \sum_{i < j} \sum_j 4\epsilon_{ij} \left[ \left( \frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left( \frac{\sigma_{ij}}{r_{ij}} \right)^6 \right] \cdot W_{VdW}(r_{ij}).$$

Кулоновская энергия взаимодействия задается потенциалом:

$$\sum_{i < j} \sum_j \frac{q_i q_j}{\epsilon \cdot r_{ij}} \cdot W_{qq}(r_{ij}).$$

В этих формулах  $W_{VdW}$  – функция переключения, гладко сводящие потенциал в ноль на больших расстояниях, а  $W_{qq}$  – экранирующая функция с радиусом  $R_q$ .

Действующие на частицы силы получают аналитически из формул для потенциалов и закладывают в соответствующие вычислительные процедуры.[4]

### 1.1.3 DPD

В данной работе рассматривается метод DPD (dissipative particle dynamics).

При моделировании систем с большим количеством частиц молекулярная динамика не может быть использована, так как потребует слишком много времени для расчета. Поэтому используется упрощенная модель вещества: группы связанных частиц описываются отдельными материальными точками с мягкими потенциалами отталкивания и с Гуковскими связями.[1]

Как и в молекулярной динамике, все частицы движутся согласно уравнениям движения Ньютона

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i, \quad \frac{d\mathbf{v}_i}{dt} = \mathbf{f}_i.$$

Для простоты, массы всех частиц принимаются равными 1, таким образом, сила, действующая на частицу равна ее ускорению. Сила складывается из трех компонент:

$$\mathbf{f}_i = \sum_{i \neq j} (\mathbf{F}_{ij}^C + \mathbf{F}_{ij}^D + \mathbf{F}_{ij}^R + \mathbf{F}_{ij}^B),$$

где сумма берется по всем парам частиц, расстояние между которыми меньше некоторого радиуса обрезаки  $r_c$ . Его можно выбрать равным  $r_c = 1$ . Консервативная сила – сила мягкого отталкивания, заданная уравнением:

$$\mathbf{F}_{ij}^C = \begin{cases} a_{ij}(1 - r_{ij})\hat{\mathbf{r}}_{ij}, & r_{ij} < 1 \\ 0, & r_{ij} \geq 1 \end{cases},$$

где  $a_{ij}$  – коэффициент отталкивания между частицей  $i$  и частицей  $j$ ; и  $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ ,  $r_{ij} = |\mathbf{r}_{ij}|$ ,  $\hat{\mathbf{r}}_{ij} = \frac{\mathbf{r}_{ij}}{|\mathbf{r}_{ij}|}$ . Через  $\mathbf{F}_{ij}^B$  обозначена сила взаимодействия между частицами с помощью валентных связей.

Если между двумя частицами есть связь, то на них действует сила

$$\mathbf{F}_{ij}^B = k(l - r_{ij}),$$

где  $k$  – коэффициент жесткости связи;  $l$  – равновесная длина связи.

Оставшиеся две силы – это диссипативная сила и случайная сила. Они задаются как

$$\mathbf{F}_{ij}^D = -\gamma w^D(r_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij})\hat{\mathbf{r}}_{ij}, \quad \mathbf{F}_{ij}^R = \sigma w^R(r_{ij})\theta_{ij}\hat{\mathbf{r}}_{ij},$$

где  $w^D$  и  $w^R$  – функции, зависящие от  $r$ , обращающиеся в нуль при  $r \geq 1$ ,  $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ , а  $\theta_{ij}(t)$  – равномерно распределенная случайная величина из промежутка  $[-1; 1]$ . Все три компоненты силы действуют вдоль прямой, соединяющей пару частиц, сохраняют импульс и момент импульса. Случайная величина вычисляется независимо для каждой пары частиц.

Как было показано[3], одна из функций  $w^D$  и  $w^R$  может быть выбрана произвольно, и этот выбор определяет вторую из них. Так же, есть связь между амплитудами сил и  $k_B T$ , т.к. сумма случайной и диссипативной сил работают термостатом в системе. В итоге, имеем:

$$w^D(r) = [w^R(r)]^2, \quad \sigma^2 = 2\gamma k_B T$$

В качестве простого выбора,

$$w^D(r) = [w^R(r)]^2 = \begin{cases} (1-r)^2, & r < 1 \\ 0, & r \geq 1 \end{cases} .$$

Получена система частиц с уравнениями для сил. Ее можно интегрировать по времени, чтобы получить динамическую модель.

## 1.2 Метод интегрирования

В данной работе используется метод численного интегрирования leap-frog, предназначенный для моделирования систем многих тел с консервативными силами. Этот метод также может быть применен в DPD[1]. Leapfrog - это модифицированная версия алгоритма Верле, состоящая из следующих уравнений:

$$\mathbf{v}_i(t + \Delta t/2) = \mathbf{v}_i(t - \Delta t/2) + \mathbf{f}_i(t)\Delta t,$$

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t + \Delta t/2)\Delta t.$$

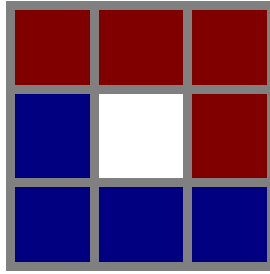
Скорости и координаты находятся со сдвигом на полшага. По координатам в момент времени  $t$  находятся силы  $\mathbf{f}_i(t)$ . Затем находятся скорости в момент  $t+0.5\Delta t$ , после чего, координаты в момент  $t + \Delta t$ . Точки расчета координат и скоростей во времени «перескакивают» друг через друга. Отсюда и следует название «leap-frog».

## 1.3 Распараллеливание DPD

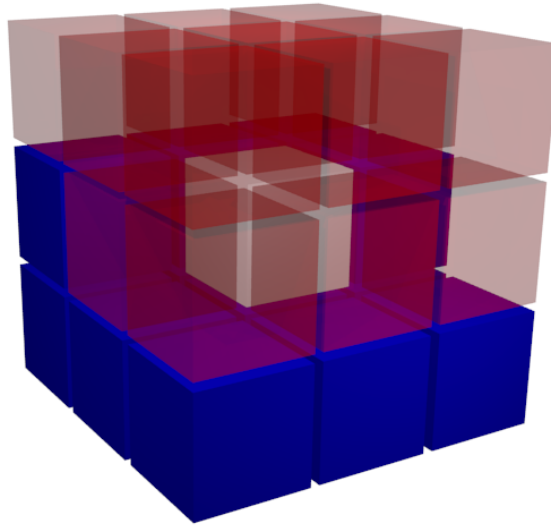
Система находится в коробке с периодическими граничными условиями. Коробка разделена на кубические ячейки с длиной стороны  $a = r_c = 1$ . При моделировании рассчитываются только взаимодействия частиц, находящихся в одной ячейке или в соседних ячейках. Остальные пары частиц находятся на расстоянии  $r > r_c$ , и учитывать их взаимодействия не нужно, т.к. сила таких взаимодействий равна нулю.

Для быстрого расчета сил при моделировании, нужно ввести структуру соседей. Выберем какую-нибудь одну ячейку, и назовем ее центральной. Все соседние ячейки делятся на два типа. Назовем их «верхними» и «нижними». В одномерном случае, у каждой ячейки было бы только две соседние ячейки: верхняя и нижняя. В двумерном случае, разделение выглядит следующим образом:





В трехмерном случае:



Здесь нижние соседи показаны синим цветом, верхние показаны красным, а центральная ячейка – белым.

Для любого разделения соседей на верхние и нижние должно выполняться условие: каждая ячейка является верхним соседом для каждого своего нижнего соседа и нижним соседом для каждого своего верхнего соседа. Для показанной выше структуры это условие выполнено.

Если центральная ячейка находится на границе коробки, то ее соседи со стороны границы находятся по другую сторону коробки. Таким образом, выполняются периодические граничные условия.

Моделирование каждого временного шага состоит из трех этапов:

1. Расчет сил
2. Интегрирование

### 3. Перераспределение частиц в ячейках

При расчете сил создается отдельный вычислительный поток для каждой ячейки. В каждой ячейке вычисляется сила взаимодействия для всех пар частиц  $i$  и  $j$  таких, что  $i \neq j$ ; частица  $i$  находится в данной ячейке, а частица  $j$  находится в данной ячейке или в одной из ее нижних соседей.

При интегрировании скоростей создается поток для каждой частицы. При интегрировании координат создается поток для каждой ячейки. Последнее необходимо для того, чтобы записать, какие частицы перешли в соседние ячейки.

На третьем этапе нужно определить, какие частицы находятся внутри каждой ячейки. Для этого используются данные предыдущего этапа о том, какие частицы переместились в другие ячейки. Данный процесс также можно распараллелить для каждой ячейки.

#### 1.4 Технические особенности работы с GPU

Основная разница между распараллеливанием на многопроцессорных системах и распараллеливанием на GPU состоит в следующем. Для максимально эффективного распараллеливания на множестве многопроцессорной системе нужно, чтобы число потоков совпадало с числом процессов. При этом, число потоков порядка  $N \sim 10 : 100$ .

GPU (graphics processing unit) может состоять из тысяч ядер. К примеру, на видеокарте Nvidia GeForce 1080 находятся 2560 ядер. Для наиболее эффективного распараллеливания на GPU нужно, чтобы число потоков  $N$  было во много раз больше числа ядер  $N_c$ :  $N \gg N_c$ .

На GPU используется блочное обращение к памяти. Ядра объединены в группы по 32. Кэш-память является общей для всех ядер одной группы. Доступ к видеопамати осуществляется каждой группой по блокам. Если данные, необходимые для работы одной группы ядер распределены в памяти хаотично, то доступ к ним будет осуществляться очень медленно. Для быстрого вычисления, все данные в памяти, используемые параллельными процессами, должны быть упорядочены. Для этого применяется несколько тактик.

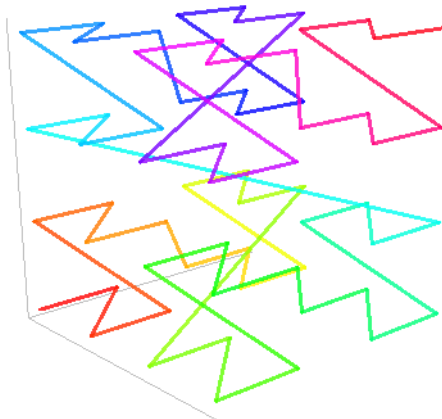
- Сортировка частиц в соответствие с номерами ячеек.

Потоки, соответствующие ячейкам, должны обращаться к памяти, в которой хранятся частицы. В следствие случайной природы моделируемого процесса, частицы расположены в ячейках хаотично и требуют много времени на сохранение в кэш.

- Сортировка номеров ячеек по кривой Пеано.

Потоки, соответствующие ячейкам должны также работать и с соседними ячейками. Поэтому, построчное кодирование ячеек – не самое эффективное. Лучше всего отсорти-

ровать их так, чтобы ячейки, которые расположены близко друг к другу в пространстве, были расположены близко и в памяти видеокарты. Для этого, пространство параметризуется кривой Пеано. В данной работе использован ее частный случай – кривая Лебега или Z-кривая.



Также стоит отметить, что при работе программы на GPU доступ к оперативной памяти компьютера осуществляется крайне медленно. Копирование данных из оперативной памяти в видеопамять также занимает очень много времени. Поэтому, все данные следует хранить в видеопамяти и все расчеты проводить только на GPU.

## 2 Результаты

### 2.1 Моделирование известных систем

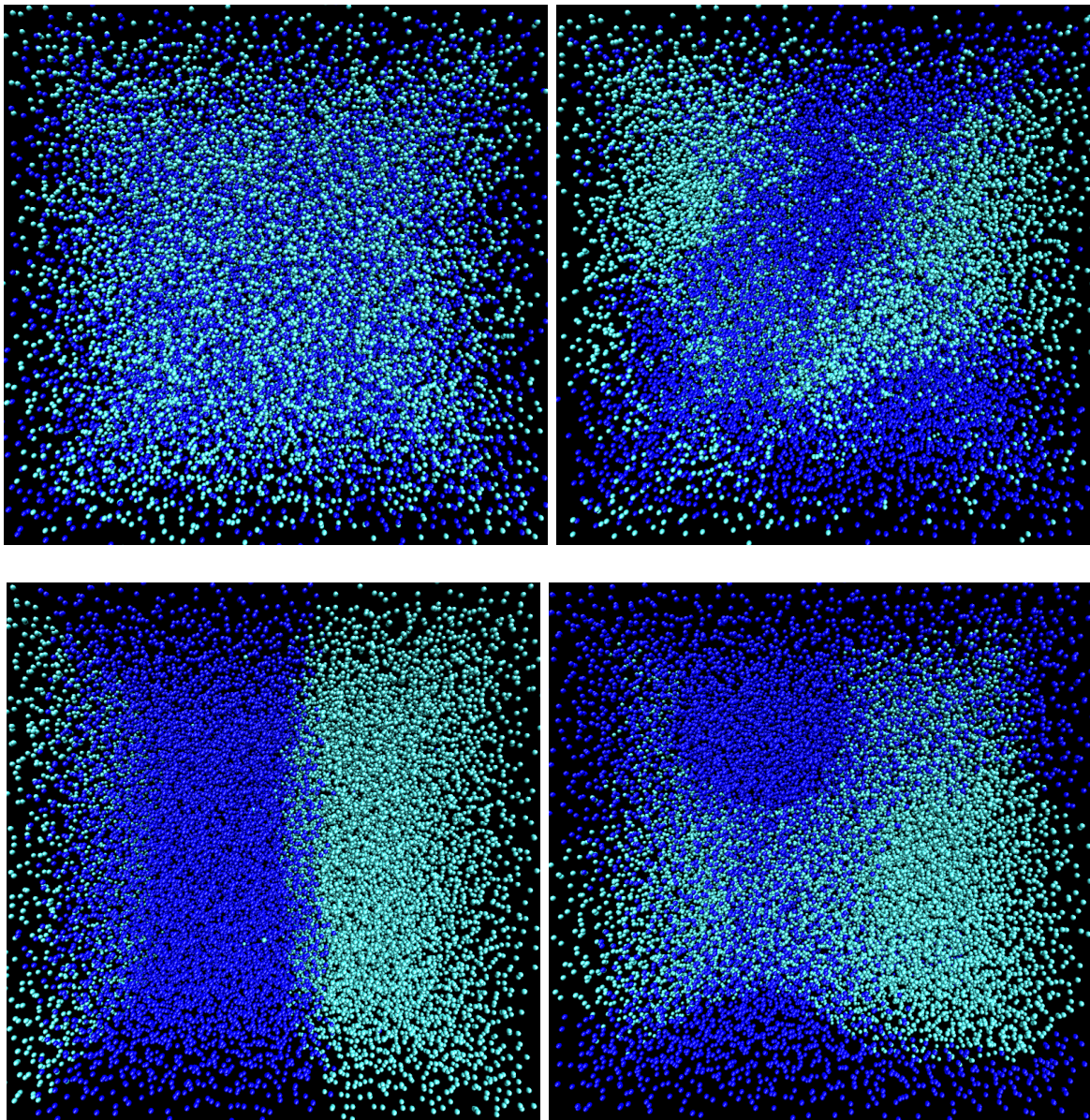
Для тестирования работы программы проводилось моделирование двух систем с хорошо известным поведением: смесь двух мономерных жидкостей и гауссов клубок.

#### 2.1.1 Расслоение мономеров

Жидкие системы – физико-химические системы, находящиеся в жидком состоянии в определенном интервале температур при любых соотношениях компонентов. Наиболее подробно изучены двухкомпонентные, или бинарные системы. [5]

Компоненты жидких систем при данной температуре могут: смешиваться друг с другом любых соотношениях; обладать ограниченной взаимной растворимостью; быть нерастворимыми друг в друге.

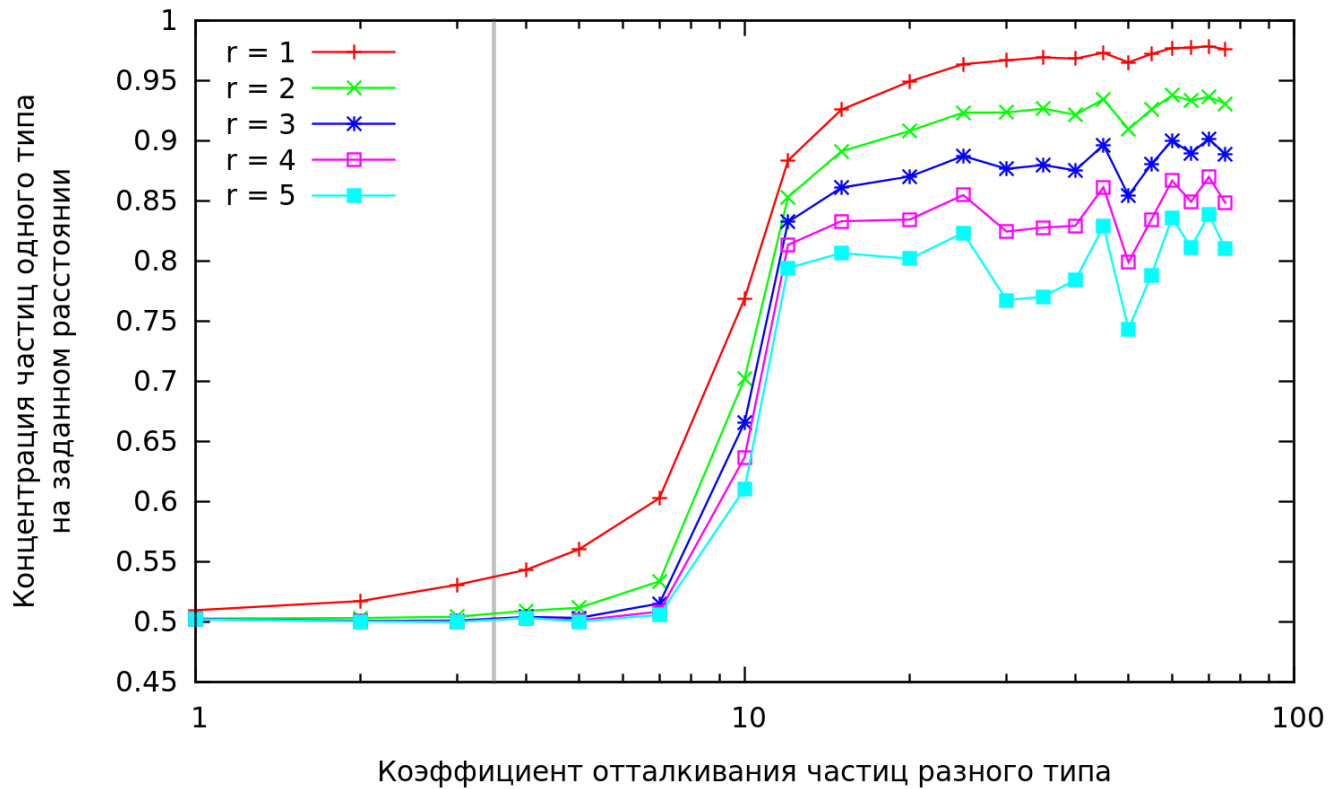
В данной работе была смоделирована двухкомпонентная жидкая система, состоящая из жидкости А и жидкости В. Коэффициент отталкивания между частицами одного вида был принят равным  $k_{AA} = k_{BB} = 25$ . Коэффициент отталкивания между частицами разного вида  $k_{AB}$  менялся в диапазоне  $[25; 100]$ . Была получена следующая картина расслоения:



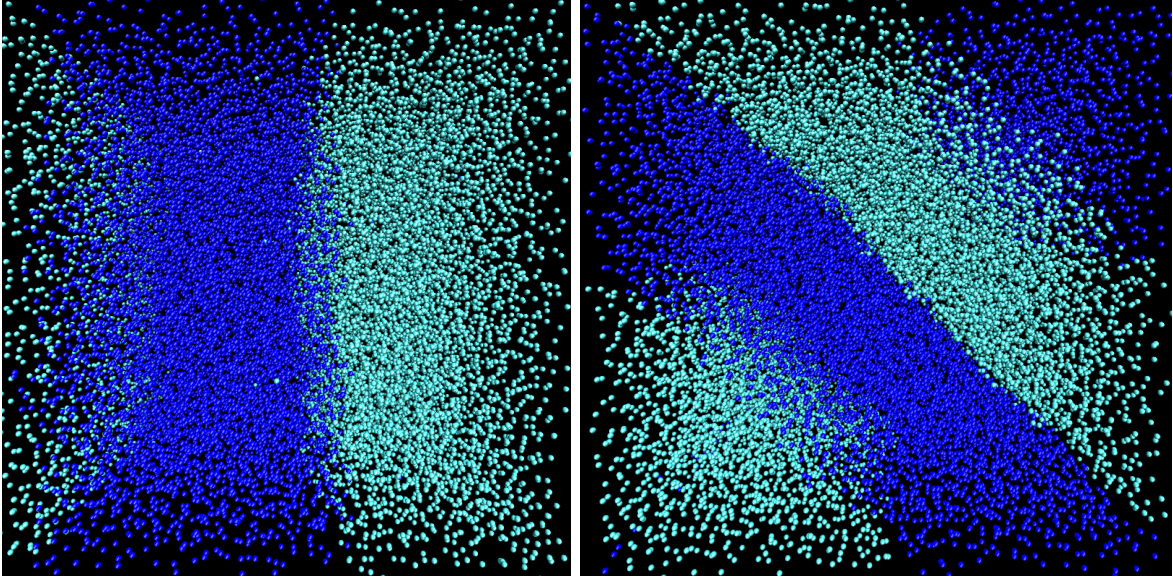
Эта картина объясняется тем, что система стремится к локальному минимуму потенциальной

энергии, а при  $k_{AB} > k_{AA} = k_{BB}$  минимуму потенциальной энергии соответствует конфигурация с наименьшей площадью поверхности раздела двух сред. Процессу расслоения противодействует броуновское движение частиц, за счет которого происходит диффузия. При значениях  $k_{AB} > 40$  образуется ламеллярная фаза. При значениях  $k_{AB} > 80$  расслоение происходит настолько быстро, что жидкость А образует шар в жидкости В, соединенный с самим собой через периодические границы коробки. Жидкость В образует такой же шар. Такое расслоение возможно только в коробке с периодическими граничными условиями и не соответствует никаким реальным наблюдениям.

Степень расслоения может быть выражена численно через отношение между числом частиц типа А, находящихся в пределах некоторого радиуса  $r$  от выбранной частицы А к полному числу частиц, находящихся в той же области.



На графике виден ярко выраженный фазовый переход от состояния смешанных жидкостей к состоянию разделенных жидкостей. Серой вертикальной линией показано теоретическое значение разности  $k_{AB} - k_{AA} = 1/0.286$ , соответствующее началу расслоения жидкой системы. Колебания графика при больших значениях  $k_{AB}$  могут быть объяснены тем, что для системы может существовать больше одного локального минимума потенциальной энергии. Пример таких разных состояний:



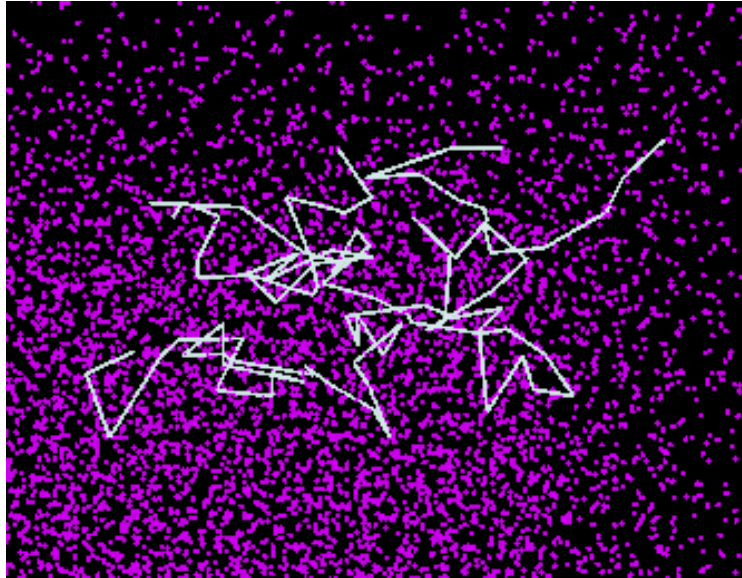
На первом изображении есть только две границы раздела общей площадью  $2a^2$ , а на втором их три, общей площадью  $2\sqrt{2}a^2$ , где  $a$  – длина стороны коробки. Система приходит в одно из этих двух состояний случайно.

### 2.1.2 Переход клубок – глобула

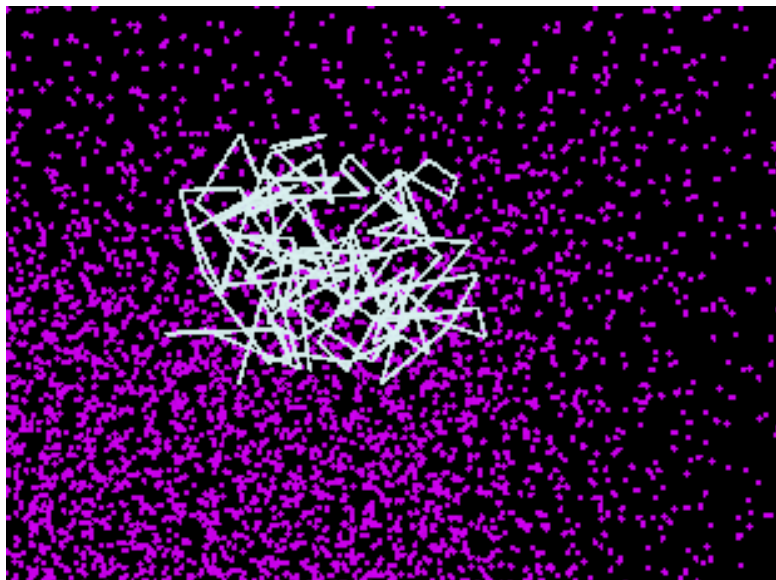
Рассмотрим полимерную цепь, находящуюся в растворителе. Персистентной длиной цепи называется длина  $d$ , такая, при длинах цепи меньше  $d$  цепь можно считать жестким стержнем, а цепь, длина которой  $L = Nd$ , может быть представлена как  $N$  свободно-сочлененных звеньев длиной  $d$ . Теория выражает значение среднеквадратичной длины цепи через  $L$  и  $d$  [6]

$$\overline{h^2} = 2d^2(L/d - 1 + e^{-L/d}).$$

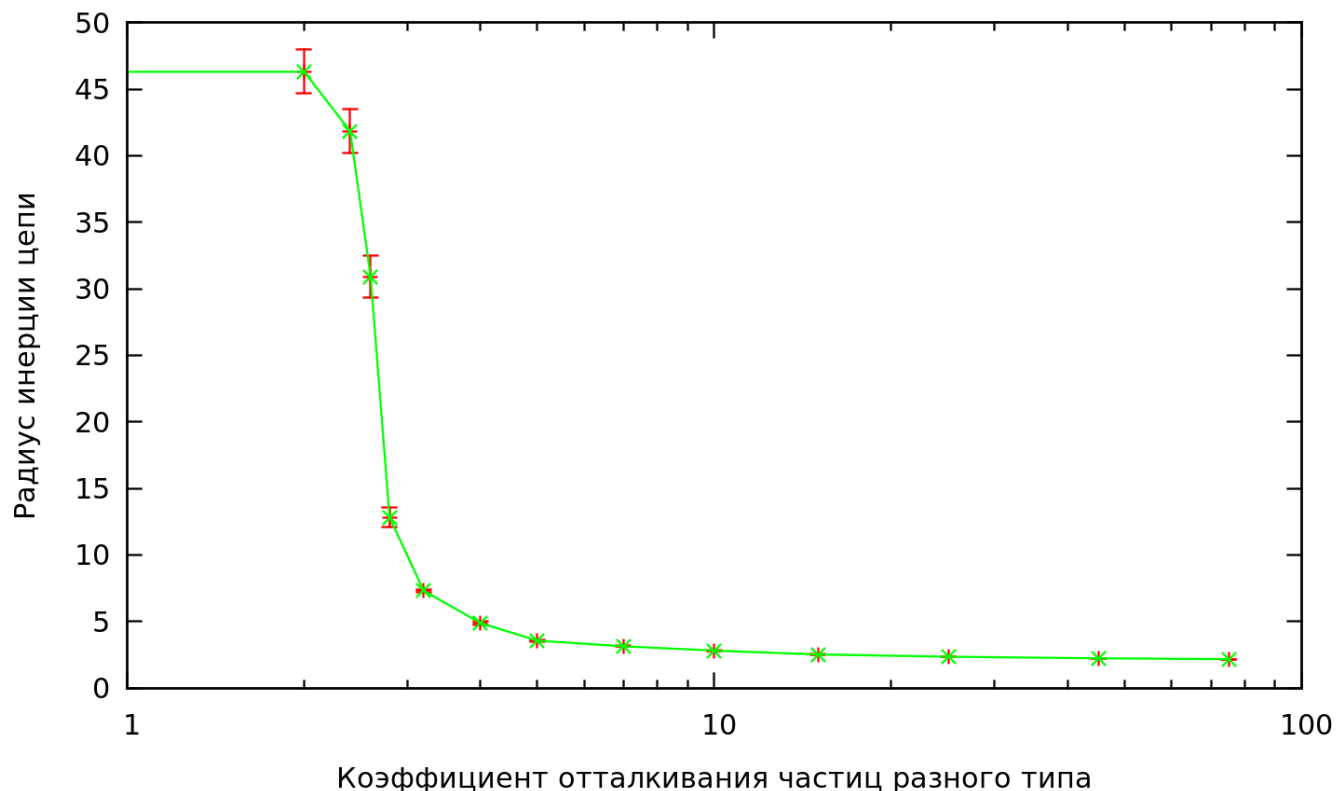
Если  $L \ll d$ , то  $\overline{h^2} = L^2$ , цепь является жесткой. Если  $L \gg d$ , то  $\overline{h^2} = 2Ld$  – цепь свернута в клубок.



Взаимодействия звеньев в клубке сходны с взаимодействиями молекул в реальном газе: сильное отталкивание на малых расстояниях и тем самым невозможность нахождения двух номеров в одном и том же месте и слабое притяжение на больших расстояниях. Существует такая температура  $\Theta$ , при которой силы притяжения и силы отталкивания компенсируют друг друга. При  $T > \Theta$  доминирует отталкивание (хороший растворитель), при  $T < \Theta$  – притяжение (плохой растворитель). Клубок раздувается или сжимается.[6] Если между звеньями реализуется сильное притяжение, то клубок схлопывается – образуется глобула.



В данной работе была промоделирована полимерная цепь из частиц типа А в растворителе из частиц типа В. Как и в предыдущем опыте, были зафиксированы коэффициенты отталкивания  $k_{AA} = k_{BB} = 25$ , а коэффициент  $k_{AB}$  изменялся в пределах [25; 100]. Для каждого значения  $k_{AB}$  был рассчитан средний радиус инерции полимерной цепи.



Полимерный клубок является рыхлым образованием: он постоянно флуктуирует. Из-за этого, радиус клубка не имеет постоянного значения, имеет смысл лишь средний радиус инерции клубка. Из графика видно, что схлопывание клубка в глобулу начинается при значении разности  $k_{AB} - k_{AA} = 2.4$ .

## 2.2 Полученное ускорение

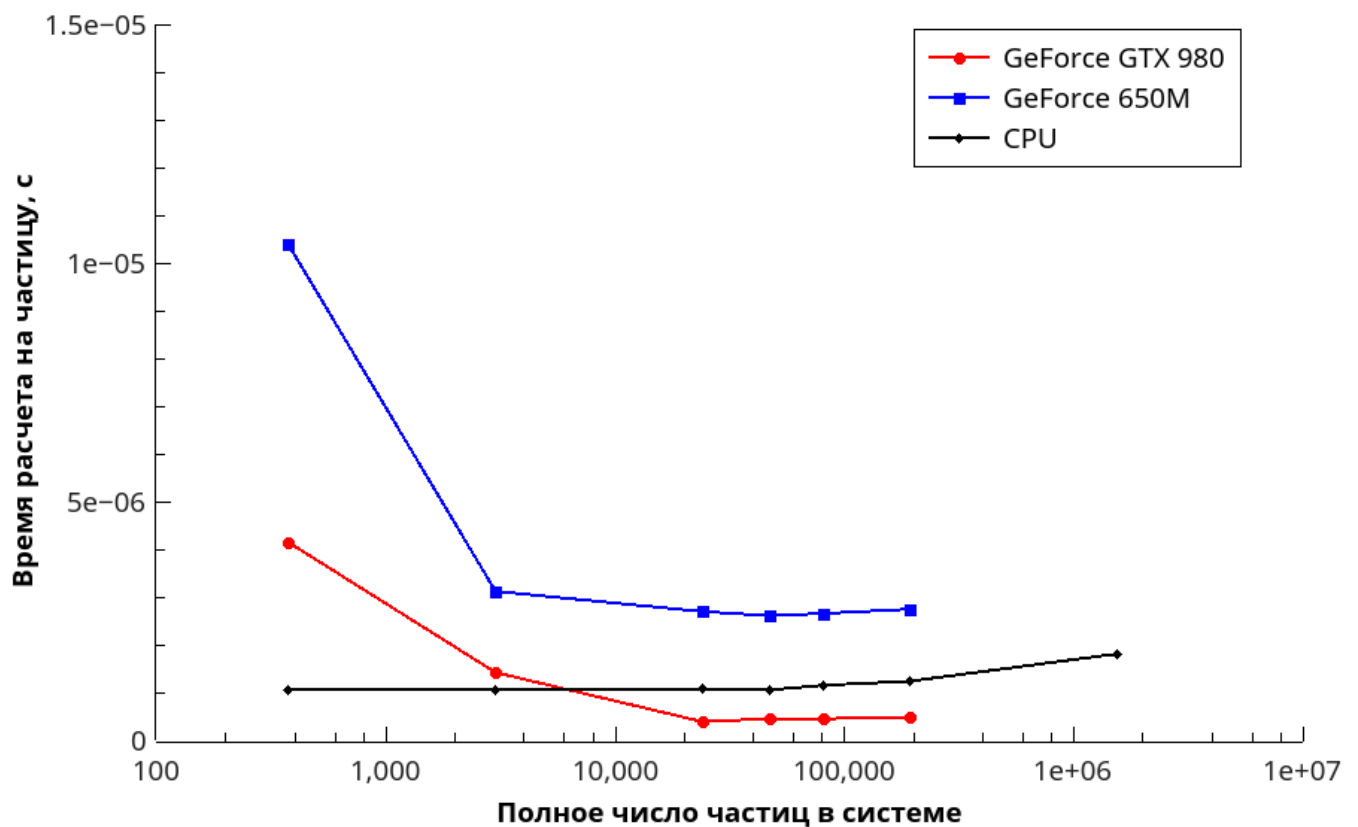
Программа тестировалась на компьютере с видеокартой Nvidia GeForce GTX 980. На этой видеокарте 2048 ядер с тактовой частотой 1126 МГц. Тактовая частота CPU в настольных компьютерах порядка 4 ГГц. Можно получить верхнюю оценку для возможного ускорения: около 500 раз.

На самом деле, эта оценка сильно завышена. Для получения более точной оценки нужно

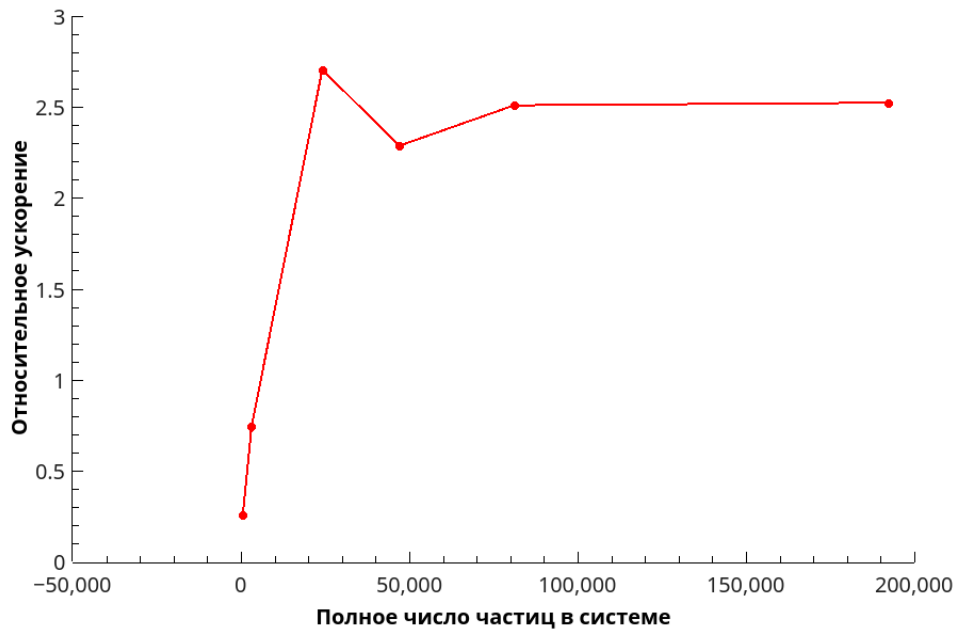


учесть более медленный доступ к памяти из-за описанной выше архитектуры графических карт. Так же, нужно помнить, что время выполнения группы параллельных процессов не может быть больше, чем время выполнения самого медленного из них. Последний эффект можно понизить, если использовать число процессов много большее, чем число ядер  $N \gg N_c$ .

В данной работе было измерено время вычисления первого из описанных примеров на трех устройствах: Nvidia GeForce 650M, Nvidia GeForce GTX 980 и CPU при разных размерах системы.



Как видно из графика, эффективность вычислений на GPU действительно возрастает при выполнении условия  $N \gg N_c$ . Так же, из графика видно, что вычисления на GPU происходят быстрее, чем на CPU. На следующем графике показано отношение скорости расчетов на GPU к скорости расчетов на CPU.



Видно, что реальное ускорение 2.5 раза, что намного меньше полученной ранее верхней оценки.

### 3 Выводы

В процессе работы было изучено поведение хорошо известных систем: гауссова цепь и смесь мономеров. Были получены навыки параллельного программирования, изучена технология Nvidia CUDA, изучена работа графических процессоров. В результате работы, была написана программа, моделирующая полимерные системы с использованием распараллеливания на GPU, которая работает в 2.5 раза быстрее, чем аналогичная программа на CPU. Достигнутое ускорение не является предельным и может быть увеличено при дальнейшей оптимизации программы.

### Список литературы

- [1] Robert D. Groot and Patrick B. Warren *Dissipative particle dynamics: Bridging the gap between atomistic and mesoscopic simulation* 1997.
- [2] А.Ю. Гросберг, А.Р. Хохлов *Физика в мире полимеров* 1989.
- [3] P. Español and P. B. Warren, *Europhys. Lett.* 30, 191 1995.
- [4] В. А. Иванов, А. Л. Рабинович, А. Р. Хохлов *Методы компьютерного моделирования для исследования полимеров и биополимеров* 2009.

- [5] И. Л. Кнунянц *Краткая химическая энциклопедия.*
- [6] М.В. Волькенштейн *Биофизика.*
- [7] Argyrios Karatrantos, Nigel Clarke, Martin Kröger *Modeling of Polymer Structure and Conformations in Polymer Nanocomposites from Atomistic to Mesoscale: A Review.*
- [8] A. A. Gavrilov, A. V. Chertovich, P. G. Khalatur, A. R. Khokhlov *Study of the Mechanisms of Filler Reinforcement in Elastomer Nanocomposites.*