

## Содержание

<b>1 Введение</b>	<b>3</b>
<b>2 Общее описание моделируемой системы</b>	<b>4</b>
<b>3 Расчёт столкновения идеальных жёстких сфер</b>	<b>5</b>
3.1 Расчёт времени до соударения двух идеальных жёстких сфер	5
3.2 Расчёт новых скоростей идеальных жёстких сфер после соударения . . . . .	5
<b>4 Начальное состояние системы. Посев</b>	<b>7</b>
4.1 Посев . . . . .	7
4.2 Расчёт коэффициента расширения системы $\beta$ . . . . .	9
<b>5 Граничные условия</b>	<b>10</b>
<b>6 Динамика моделируемой системы. События</b>	<b>15</b>
6.1 Изменение положения частиц в системе . . . . .	15
6.2 Типы событий . . . . .	15
<b>7 Собственные времена частиц и глобальное время системы</b>	<b>16</b>
7.1 Синхронизация частиц . . . . .	16
7.2 Поиск ближайшего события для частицы . . . . .	16
<b>8 Бинарное дерево как структура данных</b>	<b>17</b>
8.1 Бинарное дерево как структура данных . . . . .	17
8.2 Добавление элемента в бинарное дерево . . . . .	17
8.3 Удаление элемента из бинарного дерева . . . . .	17
8.4 Сравнение алгоритмов на основе бинарного дерева и линейного массива . . . . .	17
8.5 Бинарное дерево для сохранения информации о всех ближайших событиях в системе . . . . .	17
<b>9 Сохранение и загрузка состояния системы</b>	<b>18</b>
<b>10 Измерение относительной плотности</b>	<b>19</b>
<b>11 Получение данных о структуре системы</b>	<b>20</b>
11.1 Профиль плотности . . . . .	20
11.2 Исследование структуры выделенного слоя частиц. Разрезы .	20
<b>12 Изменение объёма системы</b>	<b>21</b>
<b>13 Измерение давления</b>	<b>22</b>
<b>14 Измерение химического потенциала</b>	<b>23</b>

<b>15 Описание управляющих команд программы</b>	<b>24</b>
<b>16 Глоссарий</b>	<b>25</b>

# 1 Введение

В данной статье приводится описание программы, позволяющей моделировать систему идеальных жёстких сфер вблизи идеальных стенок. Описывается модель жёстких сфер, модель идеальной стенки и модель используемых периодических граничных условий.

Описание, приведённое в данной статье, соответствует программе, написанной для моделирования исследуемой нами системы жёстких сфер, и может быть использовано для анализа алгоритма, реализованного в написанной программе.

Программа, реализующая описанный в этой статье алгоритм, написана на языке C++. Существует несколько версий программы, в которых реализация периодических граничных условий и некоторых функций может отличаться от описанного в этой статье алгоритма, в данной статье описывается только последний вариант реализации алгоритма, актуальный на данный момент, а так же приводится краткий обзор используемых ранее алгоритмов и периодических граничных условий с целью исторического обзора пути развития применяемых нами алгоритмов, а так же в целях объяснения причин, по которым мы выбрали имеено данные граничные условия и определённую реализацию некоторых алгоритмов (выбор используемых алгоритмов, в основном, продиктован их оптимальностью - предпочтение отдавалось алгоритмам, требующим наименьшего времени для получения точного решения).

В разработке физической модели моделируемой системы и алгоритма, описанного в данной статье и реализованного в программе на языке C++, принимали активное участие:

Вешнев Владимир Петрович

Гераськин Алексей ?

Нурлыгаянова Марина Николаевна

Нурлыгаянов Тимур Артурович

Данная статья и любая информация, описанная в ней, а так же исходный код программы, реализующий данный алгоритм, может быть использован, изменён или опубликован только с согласия всех описанных лиц.

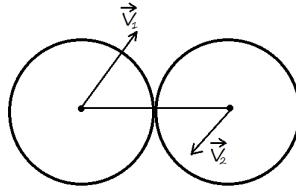
## **2   Общее описание моделируемой системы**

### 3 Рассчёт столкновения идеальных жёстких сфер

#### 3.1 Расчёт времени до соударения двух идеальных жёстких сфер

#### 3.2 Расчёт новых скоростей идеальных жёстких сфер после соударения

Рассмотрим общий случай соударения двух частиц, представляющих из себя идеальные жёсткие сферы:



Здесь  $\vec{v}_1$  - скорость первой частицы,  $\vec{v}_2$  - скорость второй частицы.

Распишем скорость  $\vec{v}_1$  как сумму  $\vec{v}_{1\perp}$  и  $\vec{v}_{1\parallel}$  и скорость  $\vec{v}_2$  как сумму  $\vec{v}_{2\perp}$  и  $\vec{v}_{2\parallel}$ , где  $\vec{v}_{1\perp}$  и  $\vec{v}_{2\perp}$  составляющие скоростей  $\vec{v}_1$  и  $\vec{v}_2$ , перпендикулярные линии, соединяющей центры сталкивающихся частиц,  $\vec{v}_{1\parallel}$  и  $\vec{v}_{2\parallel}$  - составляющие скоростей  $\vec{v}_1$  и  $\vec{v}_2$ , параллельные линии, соединяющей центры частиц.

При соударении этих частиц должен выполняться закон сохранения энергии и закон сохранения импульса. Закон сохранения импульса для рассматриваемой системы двух частиц в скалярном виде может быть записан как

$$\begin{cases} v_{1\perp} + v_{2\perp} = v'_{1\perp} + v'_{2\perp}, \\ v_{1\parallel} + v_{2\parallel} = v'_{1\parallel} + v'_{2\parallel}, \end{cases} \quad (1)$$

где  $v'_{1\perp}$ ,  $v'_{1\parallel}$ ,  $v'_{2\perp}$ ,  $v'_{2\parallel}$  - скорости частиц после соударения.

Запишем закон сохранения энергии для системы из двух сталкивающихся идеальных сфер:

$$\begin{cases} E_1 = v_{1\perp}^2 + v_{1\parallel}^2 + v_{2\perp}^2 + v_{2\parallel}^2, \\ E_2 = v_{1\perp}'^2 + v_{1\parallel}'^2 + v_{2\perp}'^2 + v_{2\parallel}'^2, \\ E_1 = E_2, \end{cases} \quad (2)$$

где  $E_1$  - полная кинетическая энергия системы из двух частиц перед соударением,  $E_2$  - полная кинетическая энергия системы из двух частиц после соударения. Потенциальная энергия частиц в рассматриваемой системе всегда равна нулю, т.к. в рассматриваемой системе не существует потенциальных полей дальнего действия.

Надо переделать вывод этого уравнения: Идеальные жёсткие сферы соударяются абсолютно упруго и изменение скоростей частиц произойдёт

только вдоль линии, соединяющей центры этих частиц. Исходя из этого предположения, мы можем записать:

$$\begin{cases} v_{1\perp} = v_{1\perp}, \\ v_{2\perp} = v_{2\perp}, \\ v'_{1\parallel} = v_{2\parallel}, \\ v'_{2\parallel} = v_{1\parallel}, \end{cases} \quad (3)$$

Запишем выражения для закона сохранения импульса (1) и закона сохранения энергии системы (2) с учётом новых скоростей (3), чтобы проверить соблюдение этих законов для случая, когда после соударения двух идеальных жёстких сфер их скорости меняются согласно (3).

Выражение для закона сохранения импульса с учётом (3) принимает вид:

$$\begin{cases} v_{1\perp} + v_{2\perp} = v_{1\perp} + v_{2\perp}, \\ v_{1\parallel} + v_{2\parallel} = v_{1\parallel} + v_{2\parallel}, \end{cases} \quad (4)$$

где все слагаемые сокращаются, что означает выполнение закона сохранения момента импульса для данного случая.

Выражение для закона сохранения энергии системы двух частиц с учётом (3) принимает вид:

## 4 Начальное состояние системы. Посев

### 4.1 Посев

Для начала моделирования необходимо создать  $N$  жёстких сфер в моделируемой системе таким образом, чтобы центры всех частиц находились в исследуемом объёме, частицы не проникали друг в друга, а так же чтобы начальный импульс системы и момент импульса системы по осям  $OX$ ,  $OY$  и  $OZ$  были равны нулю.

Процедура инициализации начальных параметров объёма, положений частиц и их скоростей называется "посев". В дальнейшем мы будем использовать этот термин для обозначения функции программы, которая позволяет произвести начальную инициализацию моделируемой системы.

Начальная плотность данной системы  $\eta_0$  будет определяться задаваемым параметрами объёма исследуемой системы и количеством частиц в данном объёме (т.к. радиус идеальных жёстких сфер  $R$  по умолчанию равен 1.0, мы полагаем его постоянным и одинаковым для всех частиц).

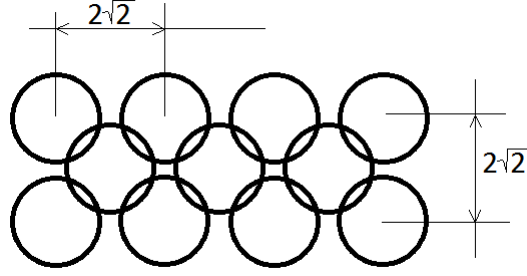
Изначально центры частиц размещаются в системе в "узлах" объёмо-центрированной кристаллической решётки, расстояние между частицами регулируется начальной плотностью системы, но при этом расстояние между двумя частицами не может быть меньше, чем два радиуса идеальных сфер (чтобы частицы не проникали друг в друга).

Расчёт начальных координат и скоростей идеальных сфер производится в несколько этапов, рассмотрим их более подробно:

1. Создаётся "кристалл" жёстких сфер с объёмо-центрированной кристаллической решёткой, содержащий  $M$  частиц. Частицы в данном кристалле касаются друг друга и образуют прямоугольный параллелепипед, рёбра которого по  $Y$  и  $Z$  равны между собой, при этом рёбра по  $X$  примерно в два раза больше рёбер по  $Y$  и  $Z$ . Это продиктовано особенностями моделируемой системы, т.к. нам необходимо задавать длину системы  $L$  в четыре раза больше, чем размер системы по  $Y$  и  $Z$ , чтобы иметь возможность исследовать свойства кристалла вблизи идеальной стенки, исключая при этом влияние второй, противоположной идеальной стенки, которая будет достаточно удалена от исследуемой части системы.

Один из углов данного параллелепипеда будет совпадать с началом координат.

Расстояние между двумя одинаковыми слоями объёмо-центрированного кристалла в случае плотного соприкосновения слоёв равно  $2 * \sqrt{2}$ :



Координаты центра данного прямоугольного параллелепипеда можно рассчитать по формулам:

$$\begin{cases} y = \frac{r_b * (NN - 1.0) + 2.0}{2.0}, \\ z = \frac{r_b * (NN - 1.0) + 2.0}{2.0}, \\ x = \frac{r_b * (2.0 * NN - 1.0) + 2.0}{2.0}. \end{cases} \quad (5)$$

где  $r_b$  - это расстояние между одинаковыми слоями объёмо-центрированного кристалла, равное  $2 * \sqrt{2}$ ,  $NN$  - количество частиц в ребре куба по  $Y$  и  $Z$ , это число задаётся при новом "посеве" частиц (см. главу "Описание управляющих команд программы").

2. Созданный кристалл копируется несколько раз так, чтобы увеличить кристалл вдвое по трём измерениям ( $x, y, z$ ), таким образом, делается 7 копий созданного на шаге №1 кристалла, которые располагаются вплотную к первоначальному кристаллу так, чтобы частицы, находящейся на границах параллелепипедов, касались друг друга.

В результате получается кристалл жёстких сфер в форме прямоугольного параллелепипеда, один угол которого совпадает с началом координат и одна сторона которого в два раза больше двух других сторон.

3. Для частиц, находящихся в кристалле, созданном на шаге №1, случайным образом задаются скорости  $v_x$ ,  $v_y$  и  $v_z$ , при этом скорости остальных частиц в системе задаются так, чтобы суммарный импульс и момент импульса системы были равны нулю.

4. Основываясь на задаваемой плотности системы рассчитываются параметры моделируемого объёма  $A$  и  $L$ , а так же коэффициент расширения  $\beta$  (мы рассмотрим подробнее вывод коэффициента расширения системы чуть ниже), на который умножаются все координаты частиц, благодаря чему объёмо-центрированный кристалл жёстких сфер расширяется до размеров моделируемой системы и частицы перестают касаться друг друга. При этом суммарный импульс и момент импульса системы остаются равными нулю, т.к. импульсы одних частиц уравновешены импульсами других частиц, расположенных симметрично относительно центра моделируемой системы.



5. Все частицы копируются ещё раз и их координаты изменяются параллельным переносом вдоль оси  $OX$  так, чтобы в результате частицы заполнили весь объём системы, заданный из расчёта  $L = 4 * A$ . Скорости частиц не изменяются, поэтому общий импульс и момент импульса системы при этом так же не изменяются. Таким образом размер системы вдоль оси  $OX$  получается в четыре раза больше размера системы вдоль осей  $OY$  и  $OZ$ , центры идеальных сфер распределены по всему объёму моделируемой системы так, чтобы частицы не проникали друг в друга и суммарный импульс системы и момент импульса системы были равны нулю.

После проведения "посева" состояние системы сохраняется во временный файл и загружается из него. Во время загрузки сохранённого состояния системы мы рассчитываем ближайшие события для каждой частицы и начинаем расчёт динамики системы. Более подробно это описано в главе "Сохранение и загрузка состояния системы".

## 4.2 Расчёт коэффициента расширения системы $\beta$

Рассмотрим подробнее вывод коэффициента расширения системы  $\beta$ .

## 5 Граничные условия

В рассматриваемой нами системе существуют два типа граничных условий:

1. Идеальная стенка
2. Периодические граничные условия

В случае идеальной стенки идеальная жёсткая сфера соударяется с идеальной стенкой по закону абсолютно упругого удара, при этом энергия частицы не изменяется, изменяется лишь проекция скорости частицы на ось  $OX$ :

$$v'_x = -v_x, \quad (6)$$

где  $v_x$  - проекция скорости частицы на ось  $OX$  до соударения с идеальной стенкой,  $v'_x$  - проекция скорости частицы на ось  $OX$  после соударения с идеальной стенкой.

Условия соударения с идеальными стенками в рассматриваемой системе можно записать так:

$$\begin{cases} x = R, v_x < 0.0 \\ or \\ x = L - R, v_x > 0.0, \end{cases} \quad (7)$$

где  $x$  - это значение координаты частицы вдоль оси  $OX$ ,  $v_x$  - значение проекции скорости частицы на ось  $OX$ ,  $R$  - радиус идеальных сфер, равный 1.0,  $L$  - расстояние между двумя идеальными стенками. Частицы не могут находиться на расстоянии менее  $R$  от идеальной стенки и координаты всех частиц в системе в любой момент времени удовлетворяют следующим правилам:

$$\begin{cases} x \geq R \\ x \leq L - R. \end{cases} \quad (8)$$

Периодические граничные условия не дают частицам беспрепятственно покидать выделенный объём, позволяя тем самым моделировать бесконечную систему частиц, рассчитывая динамику системы, содержащую только небольшое количество частиц.

Условия, которые накладываются на частицы, находящиеся в моделируемой системе:

1. Центр каждой частицы всегда находится в системе или на её границе. Положения центров каждой частицы в системе в любой момент времени удовлетворяют следующей системе уравнений:

$$\begin{cases} x \geq R, \\ x \leq L - R, \\ 0 \leq y \leq A, \\ 0 \leq z \leq A, \end{cases} \quad (9)$$

где  $x$ ,  $y$  и  $z$  - координаты частицы в трехмерной системе координат,  $R$  - радиус частицы, представляющей из себя идеальную жесткую сферу,  $L$  - расстояние между двумя идеальными стенками и  $A$  - это расстояние между двумя противоположными плоскостями, представляющими периодические границы.

2. Частицы могут пересекать границы системы, заданные уравнениями:

$$y = 0 \quad (10)$$

$$y = A \quad (11)$$

$$z = 0 \quad (12)$$

$$z = A \quad (13)$$

И не могут пересекать границы, заданные уравнением:

$$x = 0 \quad (14)$$

$$x = L \quad (15)$$

т.к. эти границы являются идеальными стенками.

При пересечении границы в момент пересечения одной из периодических границ системы центром частицы на одной из других периодических границ создаётся частица с энергией, равной энергии данной частицы и движущаяся согласно тому же уравнению движения, что и исходная частица. Исходная частица, пересекающая границу системы, при этом удаляется.

3. Если для координат центра частицы выполняется одно из следующих условий:

$$0 \leq y \leq 1 \quad (16)$$

$$A - 1 \leq y \leq A \quad (17)$$

$$0 \leq z \leq 1 \quad (18)$$

$$A - 1 \leq z \leq A \quad (19)$$

то существует один "образ" данной частицы, который движется согласно тому же уравнению движения, что и исходная частица. При пересечении центром частицы одной из границ системы "образ" становится обычной частицей, а сама частица уничтожается. При этом энергия частицы не изменяется, меняются лишь координаты центра частицы.

4. Центр "образа" всегда находится вне системы, что позволяет не учитывать "образ" как отдельную частицу. Любой частице в системе может соответствовать либо ни одного "образа" либо только один "образ".

5. При столкновении частицы или ее "образа" с другой частицей, "образом" другой частицей или идеальной стенкой скорость частицы изменяется, как и её уравнение движения. При этом "образ" соответствующий этой частице, уничтожается (если он существовал) и создаётся новый "образ" (если

это необходимо - см. пункт №3), который будет двигаться согласно новому уравнению движения.

6. "Образ" может иметь скорость, отличную от скорости самой частицы, при этом он обязательно должен перемещаться согласно тому же уравнению движения, что и исходная частица (по той же линии скорости или по линии скорости, полученной параллельным переносом по ОХ, что будет рассмотрено отдельно). При столкновении "образа" с другой частицей или "образом" другой частицы учитывается скорость исходной частицы, а не её "образа" (при расчёте новых скоростей), что позволяет сохранять общую энергию системы постоянной.

7. При создании "образа" некоторой частицы позиция его центра выбирается так, чтобы он находился на линии скорости исходной частицы, в точке, из которой он достигнет одной из периодических границ системы в то же самое время, когда центр исходной частицы будет находиться в плоскости пересекаемой ею границы.

Рассмотрим подробнее вычисление координат центра создаваемого "образа". Уравнение движения центра сферы по прямолинейной траектории с постоянной скоростью в вектном виде можно записать так:

$$\vec{r} = \vec{r}_0 + \vec{v} * t \quad (20)$$

где  $\vec{r}$  - радиус вектор нового положения центра частицы,  $\vec{r}_0$  - радиус вектор начального положения центра частицы,  $\vec{v}$  - скорость частицы и  $t$  - время движения.

Распишем это выражение в проекциях на оси координат:

$$\begin{cases} r_x = r_{0x} + v_x * t, \\ r_y = r_{0y} + v_y * t, \\ r_z = r_{0z} + v_z * t. \end{cases} \quad (21)$$

из этой системы уравнений получаем:

$$\begin{cases} t = \frac{r_x - r_{0x}}{v_x}, \\ t = \frac{r_y - r_{0y}}{v_y}, \\ t = \frac{r_z - r_{0z}}{v_z}. \end{cases} \quad (22)$$

Полученная система уравнений (22) позволяет рассчитывать время, требующееся для перемещения центра частицы из одной позиции в другую.

Когда центр частицы находится на границе системы, прямая, описывающая уравнение её движения, пересекает плоскость, которой описана эта граница. Рассчитав времена, которые потребуются частице для того, чтобы её центр оказался в плоскости каждой границы, мы можем вычислить границу, на которой центр выбранной частицы окажется быстрее всего. Для этого необходимо решить четыре системы уравнений (одна система уравнений для каждой периодической границы):

$$\begin{cases} y = 0, \\ t = \frac{r_y - r_{0y}}{v_y}. \end{cases} \quad (23)$$

$$\begin{cases} y = A, \\ t = \frac{r_y - r_{0y}}{v_y}. \end{cases} \quad (24)$$

$$\begin{cases} z = 0, \\ t = \frac{r_z - r_{0z}}{v_z}. \end{cases} \quad (25)$$

$$\begin{cases} z = A, \\ t = \frac{r_z - r_{0z}}{v_z}. \end{cases} \quad (26)$$

полагая при решении этих систем уравнений  $r_y = y$  и  $r_z = z$ , мы получим 4 решения для  $t$ :

$$\begin{cases} t_1 = \frac{0 - r_{0y}}{v_y}, \\ t_2 = \frac{A - r_{0y}}{v_y}, \\ t_3 = \frac{0 - r_{0z}}{v_z}, \\ t_4 = \frac{A - r_{0z}}{v_z}. \end{cases} \quad (27)$$

где  $r_{0y}$ ,  $r_{0z}$  - проекции радиус вектора текущего положения частицы на оси  $OY$  и  $OZ$ .

Необходимо найти  $t_{min}$ , удовлетворяющее условиям:

$$\begin{cases} t_{min} > 0, \\ t_{min} \leq t_1, \\ t_{min} \leq t_2, \\ t_{min} \leq t_3, \\ t_{min} \leq t_4, \end{cases} \quad (28)$$

где  $t_{min}$  есть время, за которое частица достигнет ближайшей границы системы, если будет двигаться с текущей скоростью. В соответствии с условием №7 'образ', соответствующий данной частице, должен за это же время достигнуть другой периодической границы системы.

Для того, чтобы вычислить координаты центра создаваемого "образа" возьмём новые значения  $v_x$ ,  $v_y$  и  $v_z$ , равные:

$$\begin{cases} v'_x = -v_x, \\ v'_y = -v_y, \\ v'_z = -v_z. \end{cases} \quad (29)$$

и рассчитаем для них уравнения (27) и (28), подставив полученное в результате решения этих уравнений значение  $t'_{min}$  значения  $v'_x$ ,  $v'_y$  и  $v'_z$  в систему уравнений (21), получим:

$$\begin{cases} r'_x = r_{0x} + v'_x * t'_{min}, \\ r'_y = r_{0y} + v'_y * t'_{min}, \\ r'_z = r_{0z} + v'_z * t'_{min}. \end{cases} \quad (30)$$

где  $r'_x$ ,  $r'_y$  и  $r'_z$  есть значения новых координат центра частицы, которые она будет иметь после прохождения через периодическую границу.

Сложив  $t_{min}$  и  $t'_{min}$  и подставив получившееся значение  $t$  в систему уравнений (21) мы получим значения координат создаваемого образа, соответствующего данной частице:

$$\begin{cases} r''_x = r_{0x} + v'_x * (t_{min} + t'_{min}), \\ r''_y = r_{0y} + v'_y * (t_{min} + t'_{min}), \\ r''_z = r_{0z} + v'_z * (t_{min} + t'_{min}). \end{cases} \quad (31)$$

здесь  $r''_x$ ,  $r''_y$  и  $r''_z$  - координаты создаваемого образа,  $r_{0x}$ ,  $r_{0y}$  и  $r_{0z}$  - текущие координаты исходной частицы, для которой необходимо создать виртуальную частицу.

## **6 Динамика моделируемой системы. События**

### **6.1 Изменение положения частиц в системе**

### **6.2 Типы событий**

В рассматриваемой нами системе могут происходить следующие события:

1. Соударение частицы или её "образа" с идеальной стенкой.
2. Соударение частицы или её "образа" с другой частицей или "образом" другой частицы.
3. Создание "образа" для частицы, находящейся в области, близкой к периодическим граничным условиям.
4. Переход частицы из одной "ячейки" в другую.
5. Замена "образа" частицей - "перерождение" частицы при прохождении через периодические граничные условия.
6. Удаление "образа" частицы.

## **7 Собственные времёна частиц и глобальное время системы**

### **7.1 Синхронизация частиц**

### **7.2 Поиск ближайшего события для частицы**



## **8 Бинарное дерево как структура данных**

Бинарное (двоичное) дерево - структура данных, хорошо известная в информатике. Позволяет сохранять упорядоченные по ключу данные так, что операции добавления и удаления элементов из такой структуры данных имеют алгоритмическую сложность  $\log_2 N$ , в то время как удаление произвольного элемента из упорядоченного списка имеет алгоритмическую сложность  $N$ , а добавление нового элемента в упорядоченный список имеет алгоритмическую сложность  $N * \log_2 N$  (поиск позиции и вставка элемента с перестановкой всех элементов "вправо").

### **8.1 Бинарное дерево как структура данных**

### **8.2 Добавление элемента в бинарное дерево**

### **8.3 Удаление элемента из бинарного дерева**

### **8.4 Сравнение алгоритмов на основе бинарного дерева и линейного массива**

### **8.5 Бинарное дерево для сохранения информации о всех ближайших событиях в системе**

## **9 Сохранение и загрузка состояния системы**

## 10 Измерение относительной плотности

## **11 Получение данных о структуре системы**

### **11.1 Профиль плотности**

### **11.2 Исследование структуры выделенного слоя частиц. Разрезы**

## **12   Изменение объёма системы**

## **13 Измерение давления**

Данная часть алгоритмов и программы ещё не реализовано, необходимо добавить описание того, как мы будем производить измерения данных и описать реализацию алгоритма в программе.

## 14 Измерение химического потенциала

Данная часть алгоритмов и программы ещё не реализовано, необходимо добавить описание того, как мы будем производить измерения данных и описать реализацию алгоритма в программе.

## **15    Описание управляющих команд программы**



## 16 Глоссарий