

МОДЕЛИРОВАНИЕ ДОСКИ ГАЛЬТОНА НА ЭВМ

Гариева Г.И., Царькова О.М.

БФ БашГУ, г.Бирск, РБ

ggarieva@yandex.ru

В век компьютеризации и информатизации научных исследований особую роль в развитии науки приобрели так называемые моделирующие программы. Различные процессы, наблюдаемые в природе, а также эксперименты, которые ставились в лабораториях, теперь с той или иной степенью точности моделируются компьютером. Огромная заслуга этих технологий в том, что они позволяют наблюдать протекание рассматриваемых процессов и получать результаты экспериментов непосредственно за компьютером, обладают высокой степенью наглядности, а также имеют большое прикладное значение (расчет строительных конструкций, компьютерная графика, игры и др.).

Доска Гальтона - это лабораторный прибор, наглядно иллюстрирующий некоторые законы теории вероятностей, в частности распределение Максвелла. Это доска, с передней стороны прикрытая стеклом, в которую в шахматном порядке достаточно часто вбиты гвозди. Вверху над гвоздями в средней части доски помещена воронка, в которую можно сыпать песок, зёрна пшена, или другие частицы. Если бросить в воронку одну частицу, то при падении вниз она испытает множество столкновений с гвоздями и, в конце концов, упадёт на стол на определённом расстоянии от центра доски. На каком расстоянии от центра доски упадёт частица предсказать невозможно из-за множества случайных факторов, влияющих на её движение. Можно говорить лишь о вероятности отклонения частицы на то или иное расстояние. Естественно ожидать, что падение частицы в центральной части стола более вероятно, чем по краям. И действительно, если через воронку сыпать частицы непрерывно, то оказывается что в

центральной части стола, находящейся под отверстием воронки, скапливается наибольшее число частиц, а по краям доски их наоборот очень мало.

Для наглядной демонстрации функционирования доски Гальтона была разработана программа на языке программирования C#, моделирующая принцип её действия.

Рассмотрим алгоритм, который реализован в программе.

Изначально определили координаты высоты и ширины создаваемой доски Гальтона (Рис.1): левая верхняя точка с координатами (x_{min}, y_{min}) и правая нижняя точка (x_{max}, y_{max}) . Исходя из этих данных, нашли значения для остальных элементов доски (гвоздей и колонок).

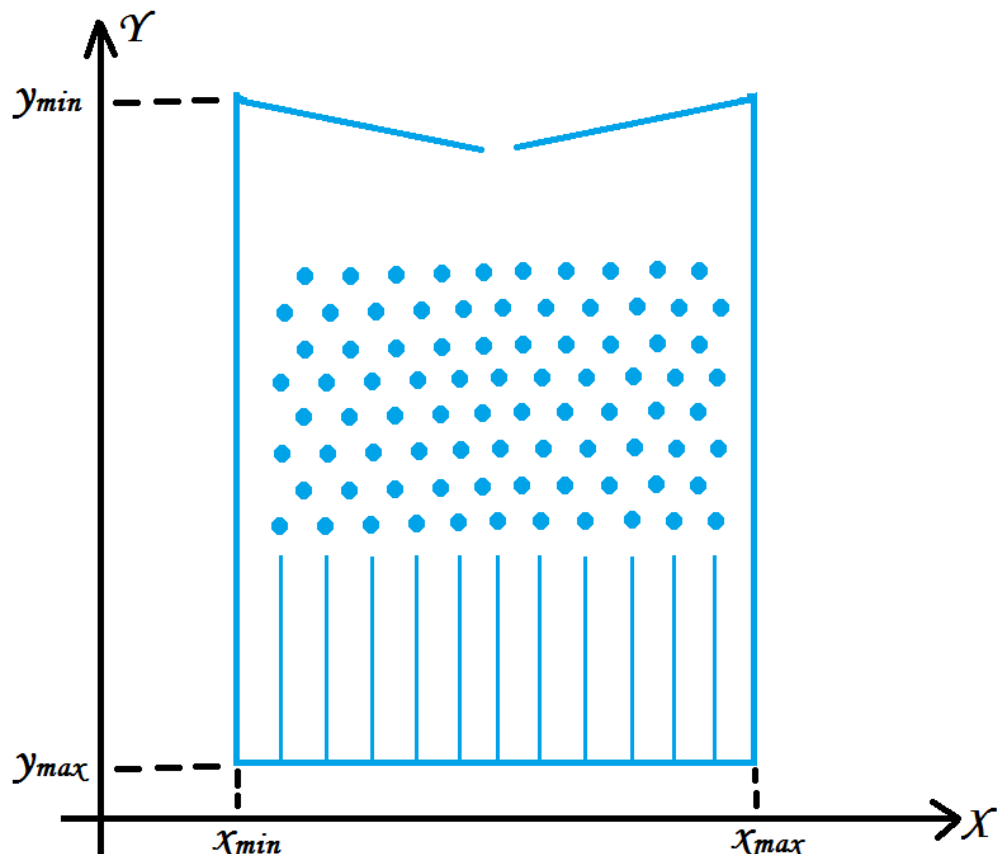


Рис. 1. Графическое отображение доски Гальтона

Падая через воронку, шарик сталкивается со средним гвоздем первого ряда, расположенным под воронкой. Далее он должен отскочить либо вправо, либо влево. Для выбора направления падения шарика используется генератор случайных чисел $\text{Random}(1)$, при обращении к которому будет получаться одно из двух чисел: 0 или 1. Если выпадает число 0, то горошина отлетит вправо; если 1 – влево (Рис.2). Таким образом, шарик проходит все 11 рядов гвоздей.

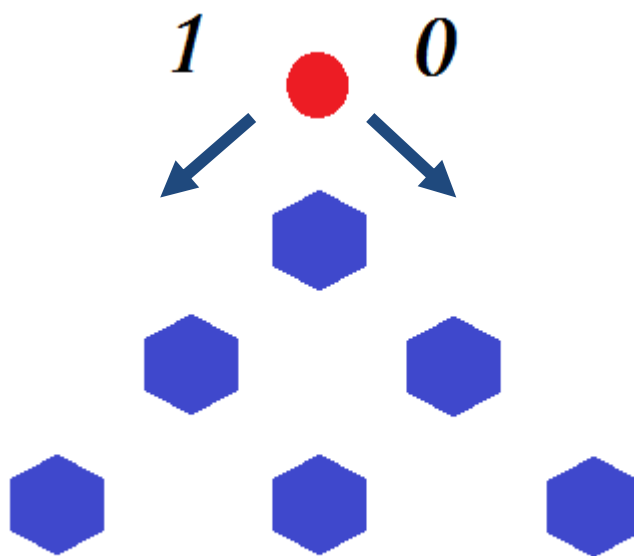


Рис.2. Движение шарика.

Количество колонок, в которые попадают горошины, заранее нами определены. Также мы знаем координаты начала и конца каждой колонки. То есть каждая колонка определяется парой координат $x1$ (начало) и $x2$ (конец). Это помогает нам определить, в какую именно колонку попала горошина и накапливать количество горошин в каждой ячейке.

На форме имеются следующие элементы (Рис.3):

1. PictureBox1 – компонент, для вывода графики;
2. Button_Go – кнопка «Старт»;
3. Button1 – кнопка «Пауза»;

4. Button2 – кнопка «Сброс»;
5. ComboBox1 – выпадающий список, для выбора количества шариков;
6. TrackBar1 – ползунок, отвечающий за скорость движения шарика;
7. Timer1 – компонент, позволяющий через определенный интервал времени обновлять форму. Запускается при нажатии на кнопки «Старт», «Сброс» и останавливается при нажатии на кнопку «Пауза».
8. Label1 – надпись «Скорость»;
9. Label2 – надпись «10...100»;
10. Label3 – надпись «Количество шаров»;
11. Label4 - надпись, для вывода количества шариков под колонками.

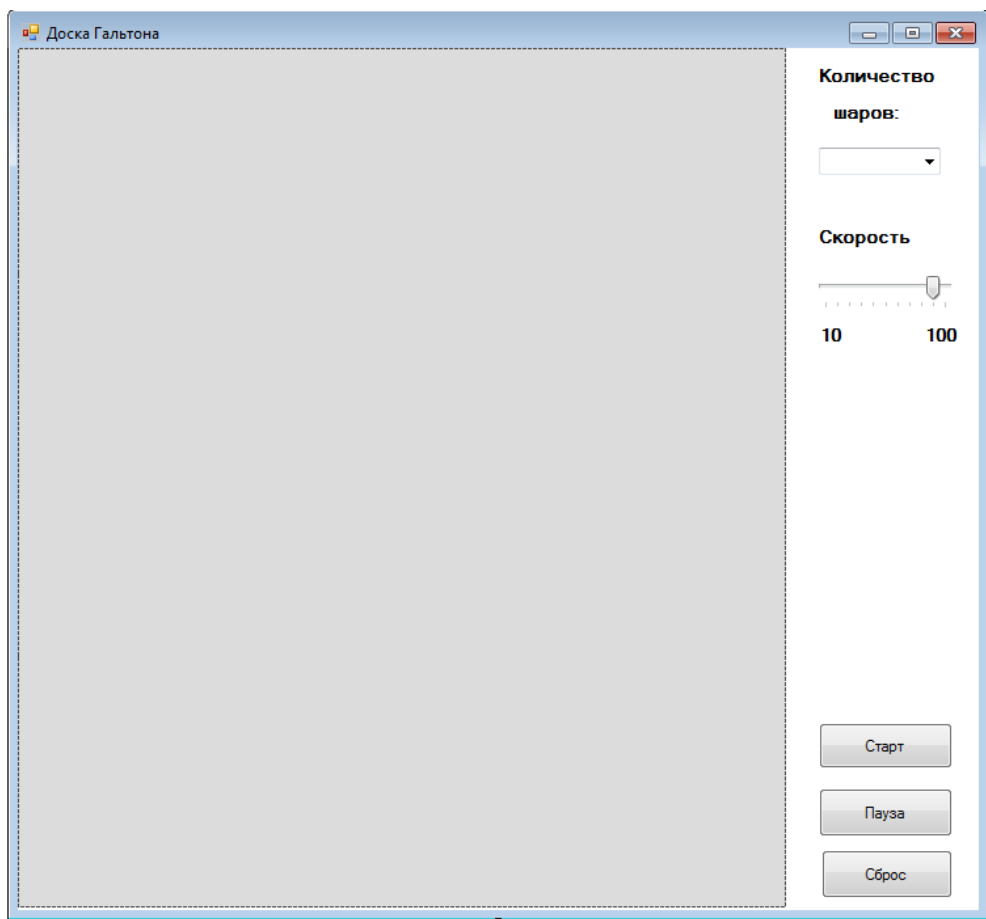


Рис.3. Форма программы.

В программе описаны следующие глобальные переменные:

- `int x = 270` - начальная горизонтальная координата шарика;
- `int y = 31` - начальная вертикальная координата шарика;

- `intd_x = 20` - шаг, на который будет меняться горизонтальная координата шарика при движении;
- `intd_y = 30` - шаг, на который будет меняться вертикальная координата шарика при движении;
- `int[] mas = newint[12]` - массив целых чисел из 12 ячеек, для хранения количества шариков в каждой колонке;
- `int[] a = newint[12]` - массив целых чисел, используемый для вывода на `PictureBox1` графического отображения шариков в каждой колонке;
- `string[] st = newstring[12]` - массив строковых элементов, используемый для форматирования надписи под колонками;
- `int k = 1` - коэффициент, с помощью которого масштабируется рисунок на форме;
- `intkol_vo = 0` - общее число шариков;

При событии `Drow` рисуется изображение доски Гальтона.

```
private void Drow(object sender, PaintEventArgs e)
{
    ...
}
```

Внутри метода описаны глобальные переменные:

`x_k = 70` – начальная горизонтальная координата;

`y_k = 70` – начальная вертикальная координата.

Гвозди рисуются с помощью метода рисования эллипса

`g.FillEllipse(br2, x_k, y_k, 10, 10)`, где ширина и высота гвоздей равны 10 пикселям, а значения `x_k` и `y_k` меняются циклически:

```
x_k = x_k + 40;
```

```
y_k = y_k + 30;
```

Колбы рисуются с помощью метода `g.DrawLine(pen1, x_k, 400, x_k, 600)`, который проводит линию, соединяющую две точки, задаваемые парами координат. Начальное значение `x_k = 75`, меняется в цикле `x_k = x_k + 40`.

Шарики на форме рисуются с помощью метода `g.FillEllipse(br, x, y, 11, 11)`.

Для того чтобы можно было проверить работу программы для большого числа горошин, используется масштабирование выводимого рисунка. То есть цена одного деления будет не одна горошина, а две, три и т.д.

На форму добавили элемент с выпадающим списком `ComboBox1` с записями 500, 1000, 1500, 2000, 2500. Здесь пользователь может выбрать, для какого числа шариков проводится испытание. Вычисляется коэффициент

```
k = Convert.ToInt32(comboBox1.SelectedItem)/500;
```

Количество горошин, уменьшенное на коэффициент k , записывается в массив:

```
a[j] = mas[j] / k;
```

Динамика изменения количества шаров, попавших в колонки, отображается в окне в виде столбчатой диаграммы. Для этого используется метод рисования прямоугольника

```
for (int j = 0; j <= 11; j++)  
{  
    g.FillRectangle(br, 36+j*40, 600 - a[j], 38, a[j]);  
}
```

При событии `timer_Tick` каждый интервал времени обновляется форма. Внутри события с помощью генератора случайных чисел выбирается одно из двух чисел 0 или 1, и записывается в переменную a :

```
int a = rand.Next(1);
```

Интервал обновления (скорость движения шарика) выбирает пользователь с помощью ползунка на форме. Это основное свойство элемента timer. Он определяется в миллисекундах. Минимальное значение для нашей программы равно 10 мс, максимальное – 1000 мс (1 сек).

Скорость движения шарика вычисляется по формуле:

$$\text{timer.Interval} = 1000 - ((\text{trackBar1.Value} - 1) * 100),$$

где `trackBar1.Value` – положение движка ползунка.

В начале пути у всех горошин одинаковые координаты (x,y) . После прохождения одного ряда гвоздей эти координаты меняются

$$y = y + d_y.$$

$$x = x - d_x \text{ или } x = x + d_x,$$

в зависимости от того вправо или влево отлетит горошина. Соответственно, после того как горошины пройдут последний ряд гвоздей у них у всех значение y будет одинаково, а значение x – разное.

Как только горошина пройдет последний ряд гвоздей, она попадает ровно в середину одной из колонок. Горизонтальную координату середины каждой колонки, зная координаты начала и конца колонок найти легко. Таким образом, мы получаем 12 чисел, сравнивая с которыми значение x , делаем вывод, в какую именно ячейку попала горошина, и увеличиваем на единицу общее число горошин и ту ячейку массива, которая соответствует данной колонке.

При запуске программы на выполнение в окне появляется изображение доски Гальтона и кнопка «Старт». При нажатии на кнопку «Старт» запускается Timer и шарики, отскакивая от гвоздей, начинают падать (Рис.4). Также становится видимым кнопка «Пауза». При нажатии на кнопку «Пауза», движение шариков останавливается и надпись на кнопке «Старт» меняется на надпись «Продолжить», становится видимым кнопка «Сброс».

При нажатии на кнопку «Сброс» обнуляются все счетчики, и обновляется форма.

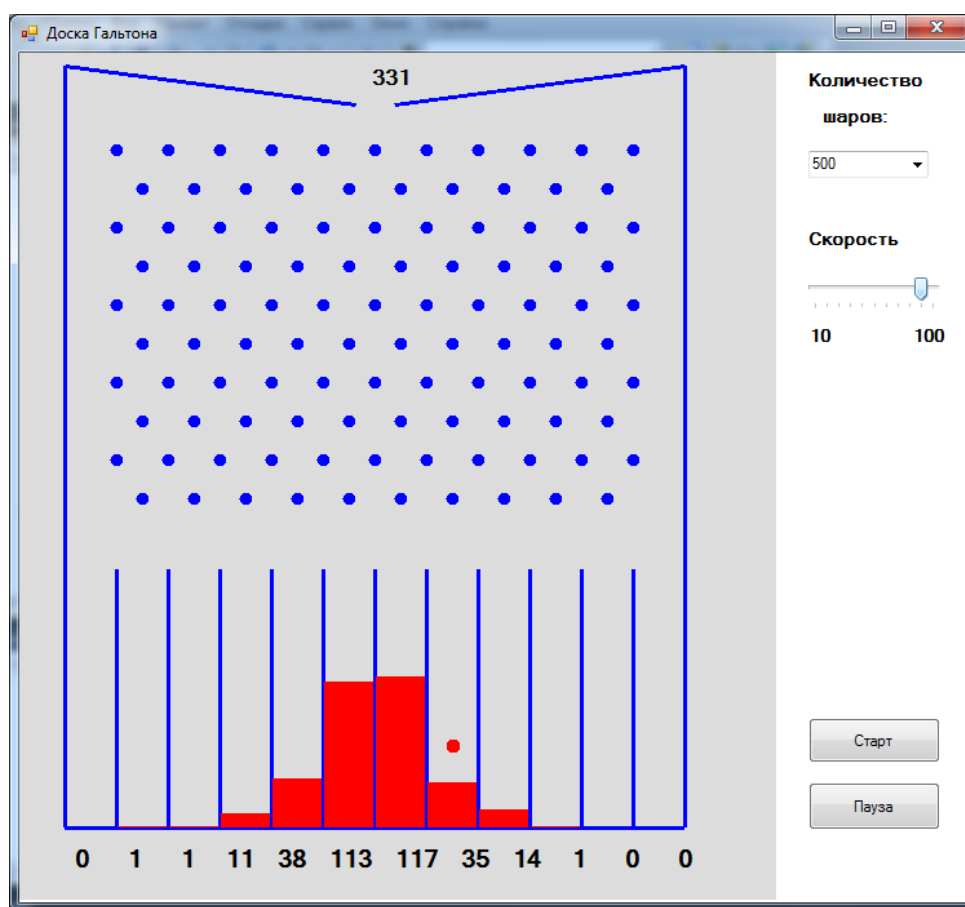


Рис. 4. Программа «Доска Гальтона»

При большом количестве частиц, прошедших через воронку, вырисовывается вполне определённая статистическая закономерность их распределения. Кривая, огибающая верхушки столбцов, имеет колоколообразную форму. Математическая функция, которой соответствует эта кривая, называется статистической функцией распределения вероятностей. В нашем конкретном случае такое статистическое распределение называют нормальным, а функцию называют функцией Гаусса.

Литература

1. Основы общей физики // Сетевой учебно-методический комплекс. URL: http://physicsbook.narod.ru/interactive/active_lk/termodinamika/max.htm (дата обращения: 15.12.2013).

