Численное интегрирование

Численное интегрирование - набор методов, позволяющих считать приближенное значение определенного интеграла.

$$\int_a^b f(x)\,dx$$

Основная идея численного интегрирования состоит в разбиении отрезка интегрирования на малые отрезки, обычно одинаковой длины, на которых исходная функция может быть приближена функцией конкретного вида, например, линейной или квадратичной.

Пусть [a, b] - отрезок интегрирования. Разобъем его на n равный отрезков длины h = (b - a) / n. Пусть $a = x_0, x_1, \dots, x_n = b$ - точки разбиения. Через I_k будем обозначать интеграл на отрезке $[x_k, x_{k+1}]$.

Способы приближения:

- Метод прямоугольников

$$I_k = h * f(x_k)$$

- Метод трапеций

$$I_k = \frac{1}{2} h * (f(x_k) + f(x_{k+1}))$$

- Метод Симпсона (приближение параболами)

$$I_k = \frac{1}{6}\ h * (f(x_k) + 4\ f(\frac{1}{2}(x_k + x_{k+1})) + f(x_{k+1}))$$

Метод Монте-Карло

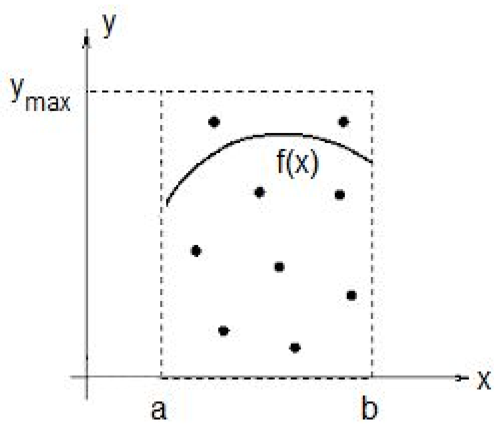
Для определения площади под графиком функции можно использовать следующий стохастический алгоритм:

- ограничим функцию прямоугольником, площадь которого можно легко вычислить как

$$(b - a) * y_{max}$$

- сгенерируем в этом прямоугольнике некоторое количество n точек, координаты которых будем выбирать случайным образом;
- определим число k точек, которые попадут под график функции;
- площадь под графиком функции можно приближенно вычислить по формуле

$$S = (b - a) * y_{max} * \frac{k}{n}$$



Два способа организовать интегрирование - обычный и быстрый.

- Способ первый (обычный)

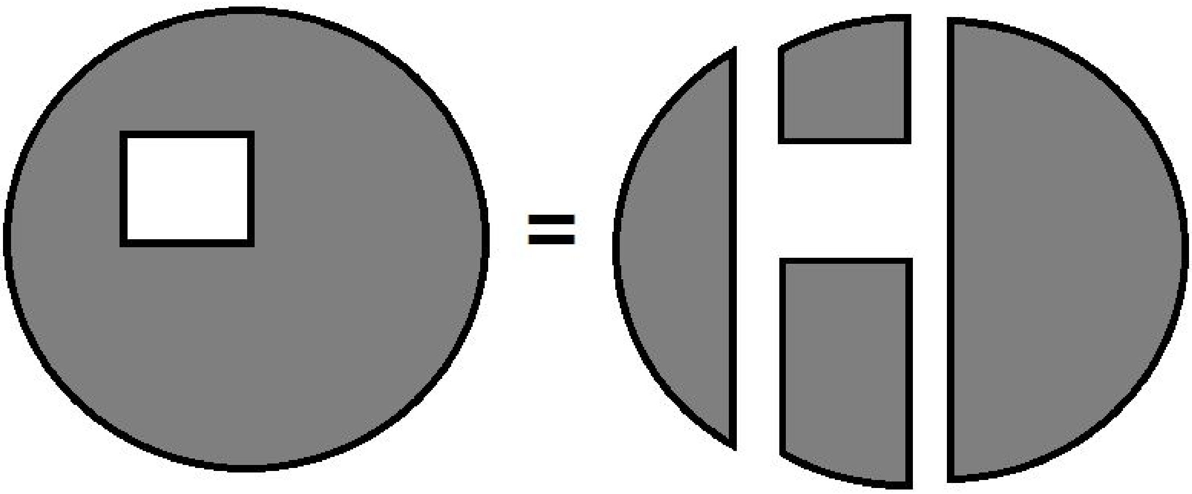
```
double trap(double be, double en) {
   return (en - be) * f((be + en) / 2);
}
double integral(double a, double b, int n) {
   double h = (b - a) / n;
   double res = 0;
   for (int i = 0; i < n; ++i)
      res += trap(be + i * h, be + (i + 1) * h);
   return res;
}
```

- Способ второй (быстрый)

```
double trap(double be, double en) {
   return (en - be) * f((be + en) / 2);
}
double integral(double be, double en) {
   double mid = (en + be) / 2;
   double a = trap(be, en);
   double b = trap(be, mid) + trap(mid, en);
   if (abs(a - b) < EPS) return b;
   else return integral(be, mid) + integral(mid, en);
}
```

Декомпозиция фигуры

Для вычисления площади фигуры сложной формы, можно представить ее в виде объединения фигур простой формы.



Numerical Integration

Numerical Integration is a family of algorithms for calculating the approximate numerical value of a definite integral. The basic problem in numerical integration is to compute an approximate solution to a definite integral to a given degree of accuracy.

$$\int_a^b f(x)\,dx$$

The key principle is to break the interval [a,b] into smaller subintervals (usually of the same length). On those subintervals the function will be approximated by a different function of a certain type: linear or quadratic.
Consider [a, b] the interval of integration. We now break it into n subintervals of equal length $h = (b - a) / n$. Consider $a = x_0, x_1, \dots, x_n = b$ - the end points of the subintervals. Consider I_k the value of the integral on the interval $[x_k, x_{k+1}]$.

Methods of approximation:

- Rectangles

$$I_k = h * f(x_k)$$

- Trapezoids

$$I_k = \frac{1}{2} h * (f(x_k) + f(x_{k+1}))$$

- Simpson's rule (parabolic approximation)

$$I_k = \frac{1}{6} h * (f(x_k) + 4 f(\tfrac{1}{2}(x_k + x_{k+1})) + f(x_{k+1}))$$

Monte Carlo methods

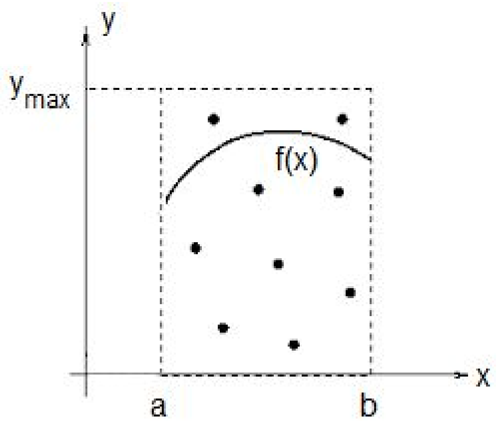
The following algorithm based on random sampling can be used to calculate the area between the function and the x-axis:

- We limit the target area within a rectangle. The rectangle's area could be easily calculated:

$$(b - a) * y_{max}$$

- We generale n points inside the rectangle. The coordinates of the points will be chosen randomly;
- We determine the number of points (k) which are located below the function curve;
- We calculate the area between the function and the x-axis:

$$S = (b - a) * y_{max} * \frac{k}{n}$$



2 algorithms of integration - the normal algo and the fast algo.

- The normal algo

```
double trap(double be, double en) {
   return (en - be) * f((be + en) / 2);
}
double integral(double a, double b, int n) {
   double h = (b - a) / n;
   double res = 0;
   for (int i = 0; i < n; ++i)
      res += trap(be + i * h, be + (i + 1) * h);
   return res;
}
```

- The fast algo

```
double trap(double be, double en) {
   return (en - be) * f((be + en) / 2);
}
double integral(double be, double en) {
   double mid = (en + be) / 2;
   double a = trap(be, en);
   double b = trap(be, mid) + trap(mid, en);
   if (abs(a - b) < EPS) return b;
   else return integral(be, mid) + integral(mid, en);
}
```

Decomposition of the figure

To calculate the area of a figure of a complex shape, you can represent it in the form of a combination of simple shapes.