Отчёт по задаче А1: метод Монте–Карло

id: 349232586
github: https://github.com/kurokolover/algorithms_hm

1 Точное значение площади

Для заданных окружностей точная площадь пересечения выражается формулой:

$$S_{\text{exact}} = \frac{\pi}{4} + 1.25 \cdot \arcsin(0.8) - 1,$$

что численно даёт:

$$S_{\text{exact}} \approx 0.9445171859.$$

2 Реализация алгоритма Монте–Карло

```
#include <iomanip>
#include <iostream>
#include <random>
using namespace std;

int main() {
    ios::sync_with_stdio(false);
    cin.tie(nullptr);
    double x1, y1, r1;
    double x2, y2, r2;
    double x3, y3, r3;
    cin >> x1 >> y1 >> r1;
    cin >> x2 >> y2 >> r2;
    cin >> x3 >> y3 >> r3;

    double minX = max(max(x1 - r1, x2 - r2), x3 - r3);
    double maxX = min(min(x1 + r1, x2 + r2), x3 + r3);
    double minY = max(max(y1 - r1, y2 - r2), y3 - r3);
    double maxY = min(min(y1 + r1, y2 + r2), y3 + r3);

    const long long N = 5000000;
    mt19937_64 rng(123456);
    uniform_real_distribution<double> distX(minX, maxX);
    uniform_real_distribution<double> distY(minY, maxY);
    long long inside = 0;
    double r1s=r1*r1, r2s=r2*r2, r3s=r3*r3;

    for(long long i=0;i<N;i++){
        double x=distX(rng), y=distY(rng);

        double d1=(x-x1)*(x-x1)+(y-y1)*(y-y1);
        double d2=(x-x2)*(x-x2)+(y-y2)*(y-y2);
        double d3=(x-x3)*(x-x3)+(y-y3)*(y-y3);

        if(d1<=r1s && d2<=r2s && d3<=r3s) inside++;
```

```
    }
    double areaBox=(maxX-minX)*(maxY-minY);
    double S = areaBox * (double(inside) / double(N));
    cout<<fixed<<setprecision(10)<<S;
}
```

3 Графики

3.1 Оценка площади от числа точек

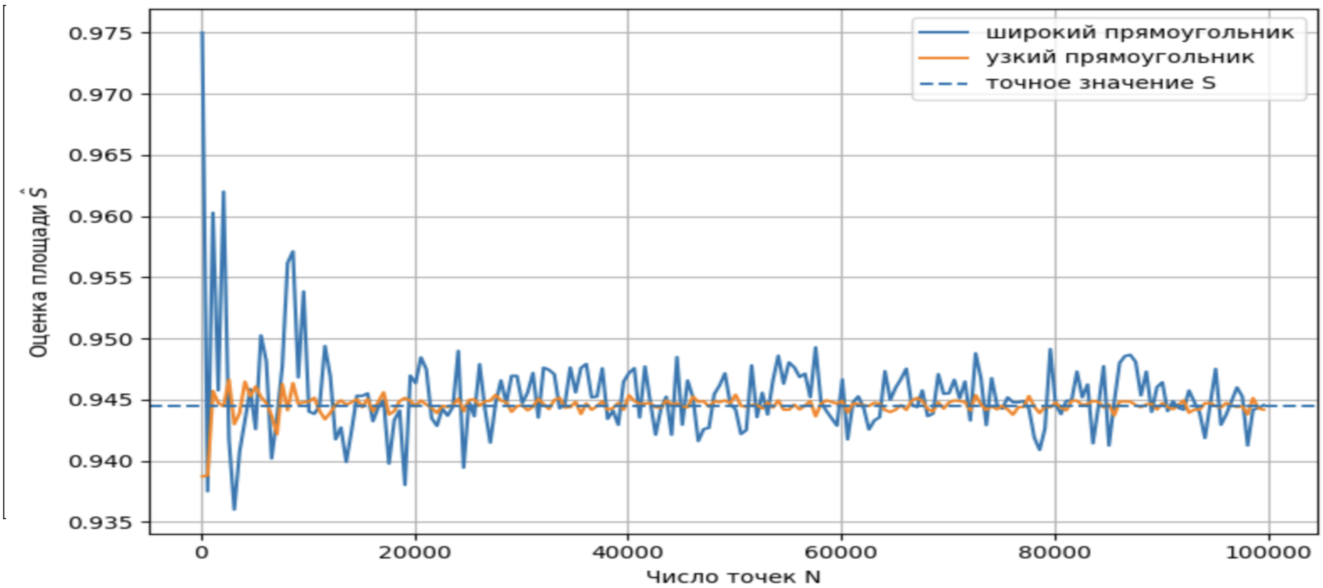

График 1: Сходимость оценки площади $\hat{S}$ при увеличении числа точек N.

3.2 Относительная погрешность

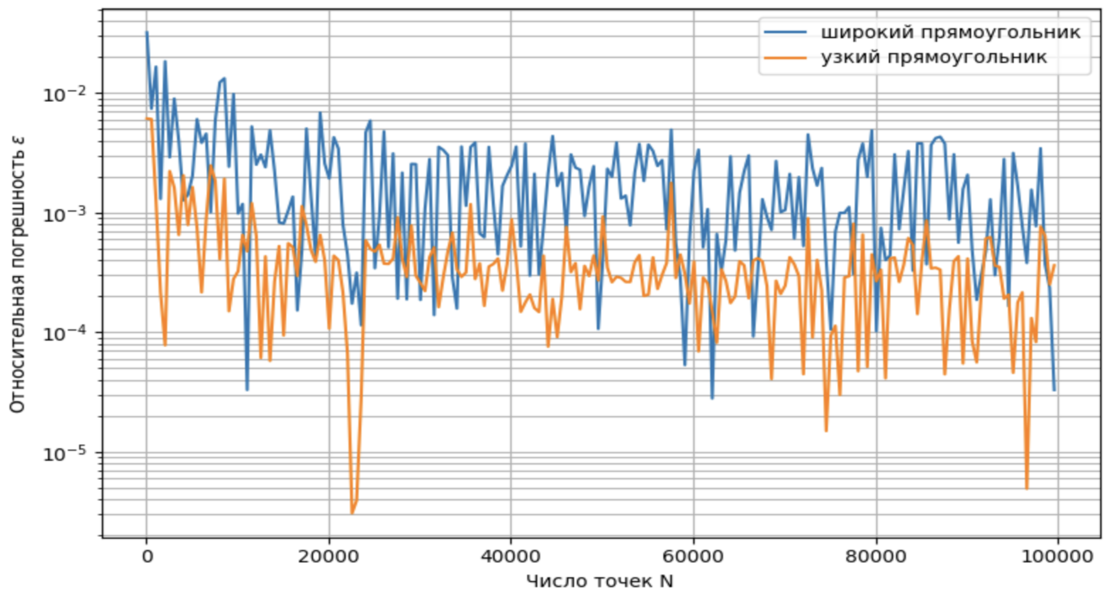

Рис. 2: Зависимость относительной погрешности ε от числа точек N.

4 Анализ результатов

- Метод Монте–Карло показывает порядка 10^{-2} при малых N.

- Использование узкой прямоугольной области даёт заметно меньшую дисперсию.
- При $N > 20\,000$ ошибка стабильно опускается ниже 10^{-3}.
- Приближение становится практически устойчивым начиная с $N \approx 50\,000$.
- В обоих случаях наблюдается характерная зависимость $O(1/\sqrt{N})$.

5 Выводы

1. Алгоритм Монте–Карло успешно оценивает площадь пересечения трёх окружностей.
2. Узкая область генерации точек даёт более точные результаты при одинаковых N.
3. Точность стабилизируется начиная с $N \approx 50\,000$.
4. Результаты хорошо согласуются с теоретической скоростью сходимости $1/\sqrt{N}$.