Univerza v Ljubljani
Naravoslovnotehniška fakulteta

Numerična integracija po metodi Monte Carlo

Poročilo pri predmetu računalniški praktikum

Izdelal: Matej Zupančič
Vpisna številka: 29010349
e-naslov: mz5400@student.uni-lj.si
Predmet: računalniški praktikum 2021/2022
Mentor: prof. dr. Goran Kugler

Doslovče, 10. 6. 2022

Kazalo:

1. Kratek opis metode

Metode Monte Carlo so skupina računalniških metod, ki uporabljajo večje število naključnih števil za natančno aproksimacijo rezultata. V projektni nalogi, katerega poročila gledate, je bil primer te metode uporabljen za računanje enodimenzionalnih integralov na tako imenovan neutežen način.

Numerično vrednost določenega integrala $I = \int_a^b f(x)dx$ najpogosteje izračunamo tako, da integracijski interval $[a, b]$ razdelimo na n enakih podintervalov, kjer meje med njimi označimo z x_i , $i \in [0, n]$. Pri tem velja $x_0 = a$ in $x_n = b$. Vrednost integrala potem lahko ocenimo kot vsoto ploščin nastalih pravokotnikov; ker je ploščina enega pravokotnika $f(x_i)\frac{b-a}{n}$, je vrednost integrala približno $\frac{b-a}{n}\sum_{i=1}^{n} f(x_i)$.

Pri neuteženi Monte Carlo integraciji pa naključno izberemo n mej med podintervali na $[a, b]$ po enakomerni porazdelitvi. Vrednost integrala je tako $\int_a^b f(x)dx = \frac{b-a}{n}\sum_{i=1}^{n} f(x_i) = (b - a) < f >$, pri čemer $< f >$ predstavlja povprečje vrednosti $f(x_i)$.

Pri tej metodi je standardna napaka $sn \approx (b-a)\sqrt{\frac{<f^2>-(<f>)^2}{\mathrm{n}}}$, kjer je $< f^2 > = \frac{1}{n}\sum_{j=1}^{n} f^2(x_j)$.

2. Program

```c
#include <stdio.h>
#include <time.h>
#include <math.h>
#include <stdlib.h>

double Funkcija (double x) {
    return (asinh(pow(x, 1.5) + sin(pow(x,2/3)) + cos(3*x) + 1/3))/(pow(x, 5) + 1);
}

double Integral (double *vi, int n,  double a, double b) {
    double vsota = 0;
    double vsota2 = 0;

    srand(time(NULL)); //Inicializacija faktorja naključnih števil.
    for (int i = 0; i < n; i++) {
        double u = (double)rand() / (double)(RAND_MAX + 1);
        double x = a + (b - a) * u;
```

```c
        vsota += Funkcija(x);
        vsota2 += pow(Funkcija(x), 2);
    }

    *vi = (vsota * (b - a)) / n;
    vsota /= n;
    vsota2 /= n;

    return (b - a) * sqrt((vsota2 - pow(vsota, 2)) / n);
}

int main(void) {

    double a = 0;
    double b = 4;
    double v = 0;
    double *vi = &v;
    int n = 2;

    FILE *kazalec_na_datoteko = fopen("Datoteka.csv", "w");

    fprintf(kazalec_na_datoteko, "poskusi; vrednost; napaka\n");

    for(int i = 0; i < 16; i++) {
        double sn = Integral(vi, n, a, b);
        fprintf(kazalec_na_datoteko, "%d; %lf; %lf;\n", n, *vi, sn);
        n *= 2;
    }

    fclose(kazalec_na_datoteko);

    return 0;
}
```

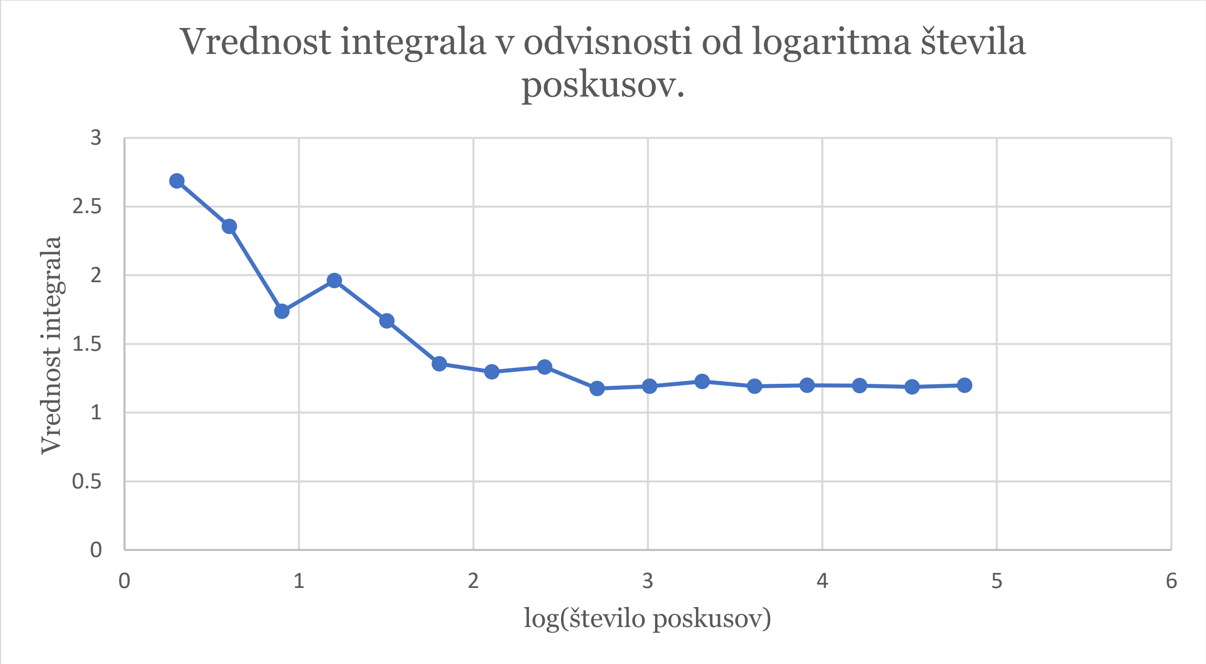
3. Rezultati

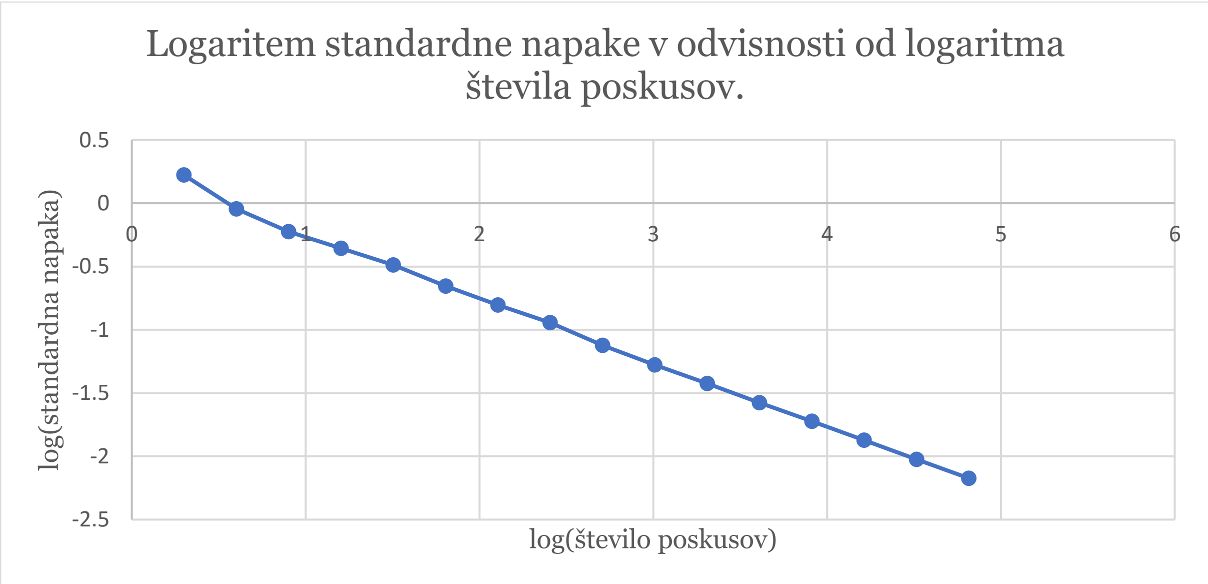
Funkcija, dana za testiranje algoritma, je bila

$$f(x) = \frac{ar\,sh\left(x^{\frac{3}{2}} + sin\left(x^{\frac{2}{3}}\right) + cos(3x) + \frac{1}{3}\right)}{x^5 + 1},$$

na intervalu $x \in [0,4]$.

Kot je vidno iz spodnjega grafa, integral f(x) lepo konvergira proti vrednosti 1,20 pri večjem številu poskusov. Prav tako se z večjim številom poskusov standardna napaka zmanjšuje, kar kaže še en graf nižje.





4. Viri in literatura

Goran Kugler: Numerična integracija po metodi Monte Carlo (2022)

Wikipedia: https://en.wikipedia.org/wiki/Monte_Carlo_integration (dostopano junija 2022)