
Filip Miletić

Računarska simulacija termodinamičkog sistema u ravni

Načinjen je računarski program za simulaciju termodinamičkog sistema u ravni, (tj. sa dva stepena slobode) metodom sukcesivne aproksimacije, a za čestice koje čine sistem pretpostavlja se da su sfernog oblika. Opisana su dva pristupa problemu: prvi, koji daje program, sporiji ali precizniji i drugi, koji postavlja više aproksimiran sistem, ali zato radi primetno brže. Program-simulator je napisan na programskom jeziku C. Rezultati rada simulatora upoređeni su sa onima koje predviđa teorija.

Uvod

U ovom radu opisaćemo jedan simulator termodinamičkog sistema. Pretpostavićemo da se čestice koje ga čine kreću bez trenja u jednoj ravni i da je svaka od njih određena masom, radijusom, položajem i brzinom. Program će raditi metodom sukcesivnih aproksimacija sa promenljivim korakom. Pošto nema nikakvog uzajamnog delovanja dvaju čestica iz sistema ukoliko one nisu u neposrednom kontaktu, računace se samo stanja sistema u vreme nekog sudara, bilo između samih čestica bilo između neke od njih i ivice ravni koja se simulira. Pošto pretpostavljamo i da je vreme sudara čestica beskrajno malo, zanemarićemo mogućnost da se u isto vreme sretnu tri ili više njih. Simulacijom dobijene podatake o stanju sistema upoređićemo sa onima koje sugeriše teorija. Smatraćemo da je simulator valjan ako su ovi rezultati saglasni.

Računarske simulacije fizičkih procesa unele su novu jednu dimenziju u istraživanja. Dobar računar i program na njemu postaju prilično korisni saradnici u svakom proučavanju neke prirodne pojave ili u eksperimentu. Ovo se naročito odnosi na eksperimente čiju bit čini posmatranje sistema od jako velikog broja čestica koje su uz to i teško dostupne, bile one udaljene (kao u astronomiji, pri proučavanju evolucije galaksija) ili sitne (kao u nekom termodinamičkom sistemu, na primer gasu). Mogućnost da se na

*Filip Miletić (1978),
Kruševac, Ratka Šakotića 39, učenik 2.
razreda Gimnazije u
Kruševcu*

računaru modelira sistem sličan onom u prirodi i da se on potom analizira u sterilnim uslovima savršene laboratorije koju predstavlja računar, može biti od velike pomoći za razumevanje osnovnih osobina tog sistema.

Da bi neki sistem uopšte mogao da se simulira na računaru, mora se prvo postaviti njegov matematički model. U praksi ovo obično nije teško. Stvaran je problem to što je vrlo verovatno da će, ako matematički nesumnjivo ispravne formule doslovce prenesemo u računarski program, njegovo izvršavanje biti presporo. Osim toga, ako sistem čini veliki broj tela, potrebna računanja mogu vrlo lako učiniti simulaciju neupotrebljivom. Zato se u pisanju simulacija nužno pribegava uprošćavanju modela, kako bi se na neki način ostvario kompromis između verodostojnosti simulacije i brzine rada programa. Koliko ćemo daleko otići u tome zavisi uglavnom od prirode problema koji rešavamo.

Opis metoda – simulatori

Simulator sa promenljivim vremenskim korakom. Simulira se kretanje čestica u ravni čije su dimenzije unapred zadate. Novi položaj čestice se računa uzv u obzir njen prethodni položaj, brzinu i vremenski interval t koji određuje koliko je vremena prošlo od prethodne iteracije. Svaku česticu određuju sledeći parametri:

- položaj (x i y koordinate)
- brzina (komponente, v_x i v_y)
- efektivni poluprečnik
- masa

Svi ovi podaci se čitaju iz datoteke na početku rada programa. U istoj datoteci se nalaze i podaci o dimenzijama površine na kojoj se simulirani sistem nalazi kao i podatak o vremenu trajanja simulacije.

U simulaciji se iterativno određuje najmanji vremenski interval posle koga će se u sistemu nešto dogoditi. Pri tome, smatramo da taj događaj može biti jedan od ovde navedenih:

Sudar čestice sa ivicom simulirane površine. Pod sudarom čestice sa ivicom simulirane površine podrazumevamo trenutak kada je rastojanje čestice od ivice jednako njenom radijusu. U tom slučaju, komponenta brzine čestice koja je upravna na ivicu sa kojom se čestica sudarila će promeniti predznak ($v_{nx} = -v_x$). Ovaj sudar je apsolutno elastičan, pa važi zakon održanja kinetičke energije i zakon održanja impulsa. Očigledno, promena znaka jedne komponente brzine čestice u potpunosti odražava stvarne efekte sudara.

Sudar dveju čestica je trenutak kada je rastojanje dveju čestica jednako zbiru njihovih radijusa. I ovaj sudar je apsolutno elastičan. Ispitivanje čestica na sudar se vrši u svakoj iteraciji i to za svaku česticu. Zamašan

broj računanja kojih ima u slučaju simulacije mnogo čestica objašnjavaju malu brzinu rada programa. Pri tome se ispituje sudar samo za čestice iz susednih ili istih oblasti.

Oblasti su uvedene da bi se simulacija unekoliko ubrzala te da bi se, koliko je moguće, linearizovala zavisnost vremena rada simulacije od broja čestica koji se simulira. Oblast je deo simulirane površine i kvadratnog je oblika. Sama simulirana površina je izdvojena na oblasti i uvek se čuva podatak o tome koje se čestice u kojoj oblasti nalaze. Dimenzije oblasti su izabrane pogodno prema radijusu čestica i srednjoj brzini, kao $d = 10 \bar{v} + 2r$, gde je \bar{v} srednja brzina, a r radijus čestica. Vreme je uštedeno utoliko što se pri ispitivanju na sudar pretpostavlja da se mogu sudariti samo čestice iz susednih oblasti.

Sudar čestica se utvrđuje na sledeći način: Neka su čestice opisane sledećim parametrima: x_{01} , y_{01} , v_{x1} , v_{y1} , r_1 i x_{02} , y_{02} , v_{x2} , v_{y2} , r_2 . Položaj obe čestice dat je izrazima:

$$x_1 = x_{01} + v_{x1} t$$

$$y_1 = y_{01} + v_{y1} t$$

$$x_2 = x_{02} + v_{x2} t$$

$$y_2 = y_{02} + v_{y2} t$$

gde je t nezavisno promenljiva – vreme. Odredimo t tako da rastojanje između centara čestica bude jednako zbiru njihovih radijusa.

$$r_1 + r_2 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Kvadriranjem ove jednačine dobija se:

$$(r_1 + r_2)^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$$

$$(r_1 + r_2)^2 = (x_{01} + v_{x1} t - x_{02} - v_{x2} t)^2 + (y_{01} + v_{y1} t - y_{02} - v_{y2} t)^2$$

$$(r_1 + r_2)^2 = [(x_{01} - x_{02}) + (v_{x1} - v_{x2}) t]^2 + [(y_{01} - y_{02}) + (v_{y1} - v_{y2}) t]^2$$

Uvedimo smene: $S = r_1 + r_2$; $D_x = x_{01} - x_{02}$; $D_y = y_{01} - y_{02}$;

$D_{vx} = v_{x1} - v_{x2}$; $D_{vy} = v_{y1} - v_{y2}$;

Tada je:

$$S^2 = (D_x + D_{vx} t)^2 + (D_y + D_{vy} t)^2.$$

Očigledno, izraz sa desne strane ove jednačine predstavlja rastojanje među česticama u funkciji vremena. Rešavanjem jednačine po t dobićemo vreme do njihovog sudara. Ukoliko je diskriminanta jednačine negativna, njena rešenja će biti imaginarna što ukazuje da u tom slučaju sudara neće biti.

Neka su: $a = D_{vx}^2 + D_{vy}^2$; $b = 2(D_v D_{vx} + D_y D_{vy})$;
 $c = D_x^2 + D_y^2 - S^2$. Onda je:

$$a t^2 + b t + c = 0$$

$$t_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a},$$

a pošto tražimo minimalno rešenje, onda je

$$t = \frac{-b - \sqrt{b^2 - 4ac}}{2a}.$$

Da bi rešenje imalo smisla, mora biti $t > 0$. Ukoliko je i ovo ispunjeno, onda t sadrži vreme do sudara dveju čestica. Simulacija se svodi na određivanje najmanjeg vremenskog intervala do sudara ma koje dve od čestica. Pošto je interval izračunat na ovde izložen način, odrede se novi položaji svih čestica posle tog intervala. Posebno se uzima u obzir sudar čestice sa ivicom simulirane površine.

Kada se utvrdi sudar dveju čestica, brzine koje će one imati posle sudara se određuju na sledeći način: Neka se čestice sudare tako da se prava koja spaja njihove centre poklapa sa x-osom. Njihove mase su m_1 i m_2 , brzine v_{x1} , v_{y1} (obe komponente brzine prve čestice) i v_{x2} , v_{y2} . Čestice se sudaraju tako da se mogu promeniti samo komponente njihovih brzina paralelnih x-osi. Odredimo u tom slučaju nove vrednosti komponenta brzina obe čestice. Kako je sudar apsolutno elastičan, to važe i zakon održanja kinetičke energije i zakon održanja impulsa pa možemo napisati:

$$m_1 v_{x1} + m_2 v_{x2} = m_1 v'_{x1} + m_2 v'_{x2}$$

$$\frac{m_1 v_{x1}^2}{2} + \frac{m_2 v_{x2}^2}{2} = \frac{m_1 v_{x1}'^2}{2} + \frac{m_2 v_{x2}'^2}{2}$$

a rešavanjem ovog sistema jednačina po v'_{x1} i v'_{x2} dobijaju se nove brzine čestica. Rešenja su:

$$v'_{x1} = \frac{v_{x1}(k-1) + 2v_{x2}}{k+1}$$

$$v'_{x2} = \frac{v_{x1}\left(\frac{1}{k}-1\right) + 2v_{x2}}{\frac{1}{k}+1}, \text{ gde je}$$

$$k = \frac{m_1}{m_2} .$$

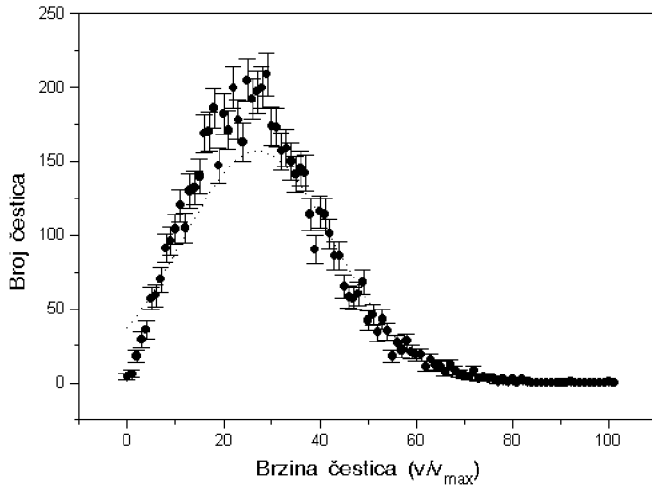
Simulator sa fiksnim vremenskim korakom. Iako je simulacija prilično verna, pokazalo se da je simulator koji koristi prethodni metod jednostavno prespor za iole složene zadatke. Vreme potrebno za simulaciju neprihvatljivo raste sa porastom broja čestica. Da bismo dobili program koji može da se nosi sa većim brojem čestica (više hiljada), moramo uvesti još jednu aproksimaciju – fiksiranje najmanjeg vremenskog intervala u kome se simulira. Pojednostavljenje je u tom smislu da se sada pomeraj čestice uvek računa tako da je vremenski interval u kome se čestica pomera stalan i unapred zadat. Tako se smanjuje broj računanja u svakoj iteraciji i smanjuje njihova složenost. Pojavljuju se, međutim i problemi – sada je moguć veći broj sudara čestica za vreme jedne iteracije; napokon, može se desiti i da čestice pri sudaru „uđu” jedna u drugu pa da rastojanje između njih u momentu sudara bude manje nego zbir njihovih radijusa. Ovi problemi se mogu delimično prevazići smanjivanjem koraka vremenskog intervala u kome se simulira. Konceptcija simulatora sa fiksnim vremenskim korakom u osnovi je slična prethodnoj njegovoj varijanti, jer se sudari čestica računaju isto.

Rezultati i diskusija

Izloženi rezultati se svi odnose na simulator sa fiksnim vremenskim korakom koji iznosi 0.1 s. Broj čestica u sistemu koji je simuliran bio je 6000, najveći koji se u simulator može uneti.

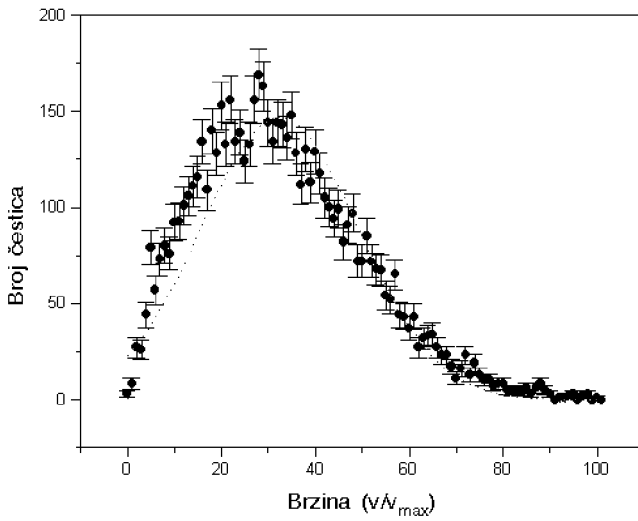
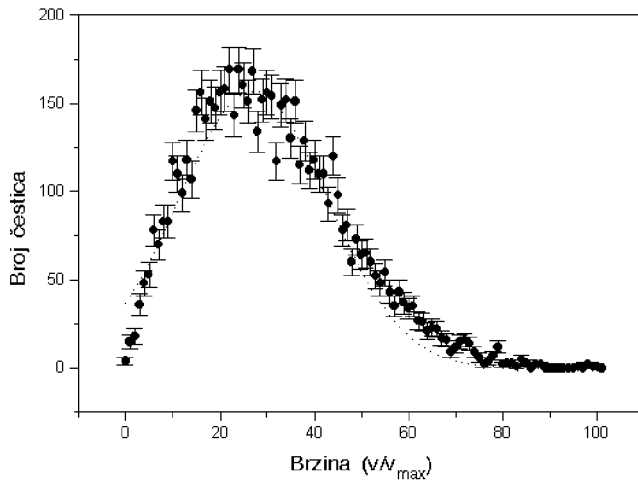
Grafici 1-3 (slika 1) prikazuju rezultate četiri eksperimenta urađena na simulatoru. Grafikon na slici 2 pokazuje broj sudara po svakoj iteraciji, a grafikon na slici 3 pokazuje ponašanje relativne greške kinetičke energije sistema. Rezultati su obrađeni programskim paketom ORIGIN.

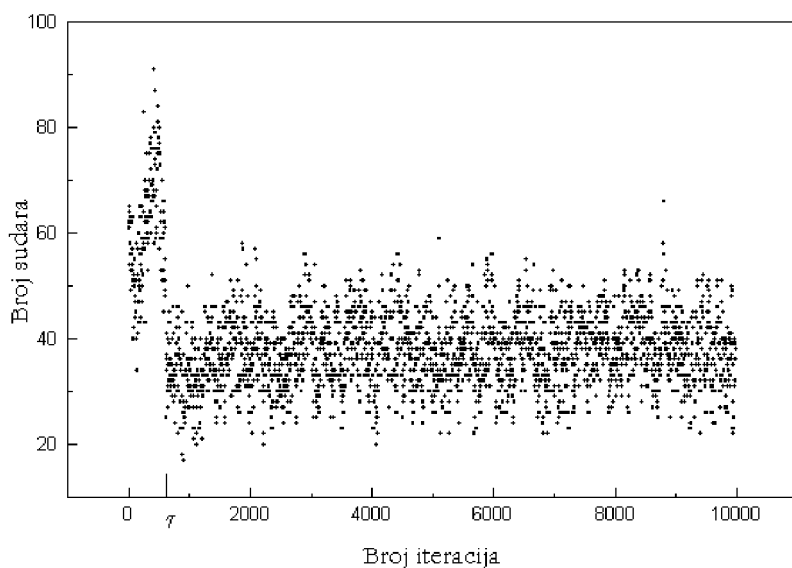
Grafici 1-3 određuju raspodelu brzina čestica u intervalu od 0 do v_{\max} . U pozadini svakog se nalazi raspodela brzina koju po teoriji treba očekivati u realnom sistemu. Sa grafika se može uočiti da se rezultati dobijeni primenom simulatora uveliko slažu sa ovim očekivanjima.



Slika 1.
Raspodela brzina čestica.

Figure 1.
Distribution of particle speed.





Slika 2.
Broj sudara čestica
po iteraciji.

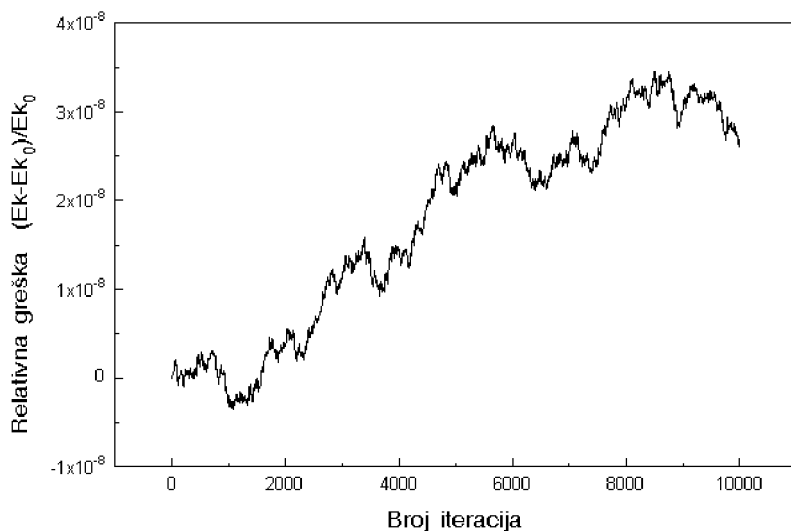
Figure 2.
Number of collisions
per iteration.

Grafikna slika 4 pokazuje broj sudara po iteraciji. Sa njega se može očitati vreme relaksacije sistema (500) koje je predstavljeno trenutkom od kada se ujednačuje broj sudara u iteracijama. Broj sudara u početku simulacije daleko premaša srednji broj sudara posle vremena relaksacije. To je zato jer su početne brzine čestica određene upotrebom generatora slučajnih brojeva sa uniformnom raspodelom, pa je početna raspodela brzina u neku ruku bila „veštačka“.

Grafik na slici 3 daje nam promenu vrednosti relativne greške kinetičke energije sistema. Ovaj parametar je usvojen kao jedan od kriterijuma koji pokazuje valjanost simulatora. Smatrali smo da možemo biti prilično sigurni u valjanost programa ukoliko greška ne pokazuje značajna kolebanja za vreme dok simulacija traje. Izvori greške su ograničena numerička tačnost svih računanja i nepreciznosti koje su prisutne zbog aproksimacija. Osobina greške simulacije je da se akumulira s vremenom. Međutim, sa grafika se može pročitati da kolebanje greške nema neki precizno definisan manir. Za sada ne možemo dati precizan odgovor zbog čega je tako.

Zaključak

Namera nam je bila da pokažemo jedan način kako se može uraditi prilično verna simulacija koja uz to može da se izvršava na, za zadatke ove prirode skromnom, personalnom računaru. Ovaj simulator može da predstavlja osnov za neki drugi program slične namene koji će podrazumevati postojanje složenijih načina interakcije čestica u sistemu. Takođe, smatramo da smo pokazali da vernija simulacija ne vodi nužno uvek rezul-



Slika 3.
Relativna greška
kinetičke energije.

Figure 3.
Relative error of
kinetic energy.

tatima koji su u praksi prihvatljivi. Za sve vreme rada podsećali smo se da je simulacija ma koje prirodne pojave je još uvek složen posao i da njena realizacija balansira stalno na oštroj granici među vernosti stvarnom modelu i postojanja prevelikim zalogajem čak i za današnje brze računare.

Literatura

- [1] Kraus L. 1993. *Programski jezik C sa rešenim zadacima*. Beograd: Mikroknjiga
- [2] Vučić V., Ivanović M. 1965. *Fizika I*. Beograd: Univerzitet u Beogradu
- [3] Benett W. R. 1976. *Scientific and Engineering problem-solving with the computer*. New Jersey: PUBLISHER

Filip Miletić

A Computer Model of 2D Thermodynamic System

A software simulation of 2D thermodynamic system in approximation of hard spheres was made. Proof of method validity is made through comparison of the simulation output to theoretical predictions. Classical example performed by the simulator is also submitted. The simulator and accompanying utilities are written in the C programming language.

