Simulacija harmonijskog i matematičkog njihala Eulerovom metodom

Fran Slijepčević

UNIRI - FIZRI

1 Uvod

U ovm ćemo radu istražiti razliku između harmonijskog i matematičkog njihala. Koristit ćemo Fortran kako bi simulirali njihala i Python za analizu podataka i izradu grafova. Koristi ćemo Eulerovu metodu za numeričko rješavanje jednadžbi gibanja, te dobivene podatke predstaviti u obliku grafova ovinosti kuta o vremenu.

2 Harmonijsko Njihalo

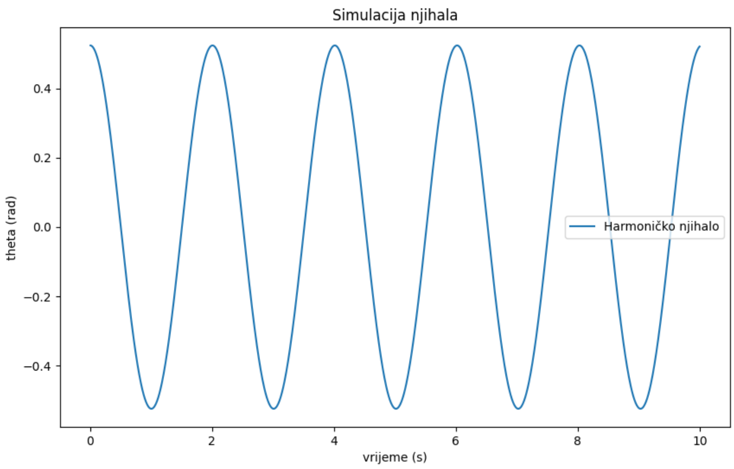
Harmonijsko njihalo model je idealiziranog njihala (oscilatora) gdje se kutni pomak θ mijenja tijekom vremena prema jednadžbi:

$$\ddot{\theta} + \frac{g}{l} \sin\theta = 0 \tag{1}$$

Harmonijsko njihalo koristi aproksimaciju malih kuteva $\sin\theta \approx \theta$ pa jednadžbu gibanja možemo svesti na :

$$\frac{d^2\theta}{dt^2} = -\frac{g}{L}\theta \tag{2}$$

2.1 Graf 1: Kutni Pomak Harmonijskog Njihala



Slika 1: Kutni Pomak Harmonijskog Njihala za početni pomak od 30 stupnjeva

3 Matematičko Njihalo

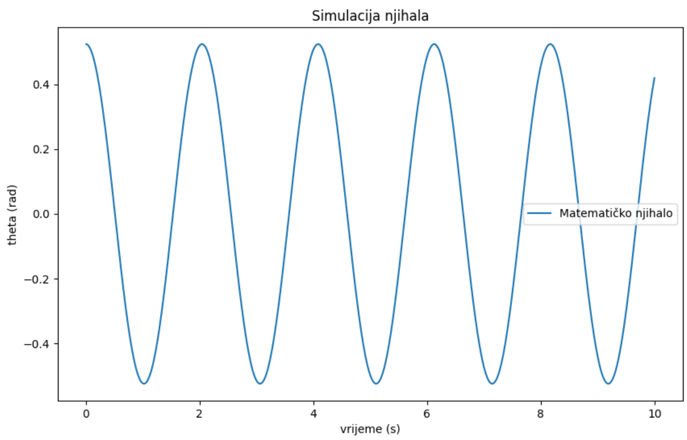
Matematičko njihalo je realističniji model koji uključuje trenje i zračni otpor. Njegova jednadžba gibanja je:

$$\ddot{\theta} + \frac{g}{l}\sin\theta + \beta\dot{\theta} = 0 \tag{1}$$

Matematičko njihalo **ne** koristi aproksimaciju malih kuteva pa se njegova jednadžba svodi na :

$$\frac{d^2\theta}{dt^2} = -\frac{g}{L}\sin(\theta) \tag{2}$$

3.1 Graf 2: Kutni Pomak Matematičkog Njihala



Slika 2: Kutni Pomak Matematičkog Njihala za početni pomak od 30 stupnjeva

Možemo primjetiti da se pozitivne amplitude nalaze točno na svake dvije sekunde kod harmonijskog dok su kod matematičkog malo pomaknute. Kako bi vizualizirali koliki je taj pomak, korisit ćemo Eulerovu metodu, simulirati oba njihala te ih prikazati na istom grafu.

4 Eulerova Metoda

Za simulaciju kretanja klatna koristili smo Eulerovu metodu, koja numerički rješava diferencijalne jednadžbe koristeći linearne aproksimacije. Eulerova metoda posebno je korisna u radi svoje jednostavnosti i učinkovistosti, što doprinosi brzini simulacije i štedi memoriju.

$$y_{n+1} = y_n + hf(t_n, y_n) \tag{1}$$

5 Simulacija pomoću Fortran-a

Koristeći Fortran, simulirali smo gibanje oba klatna za šest različitih početnih kutova, analizirajući kako se kutni pomak, brzina, i energija mijenjaju tijekom vremena.

Simulirati možemo bilo koji početni kut, no u radu ćemo predstaviti 4 reprezentativna kuta iz kojih je vidljiva razlika koja nastaje radi aproksimacije sinusa pri većim pomacima od stabline točke. Radi simetričnosti uzimamo u obzir samo kuteve od 0 do 180 stupnjeva.

6 Program

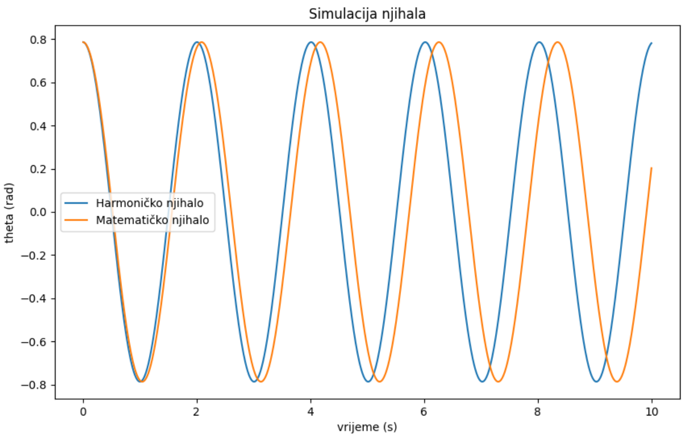
Program koji smo napisali u Fortranu omogućuje korisniku da unese početni kut njihala, te simulira ponašanje njihala kroz narednih 10 sekundi. Period od 10 sekundi smo podjelili na 1000 vremenskih koraka kroz koje prolazimo u klasičnoj do petlji. Korisniku smo omogućili unos u stupnjevima pošto je vizualizacija otklona lakša kada baratamo sa svakodnevnim jedinicama.

Nakon simulacije program će u datoteku spremiti sve vremenske korake te pripadajuće vrijednosti kuteva theta za matematičko i harmonijsko njihalo.

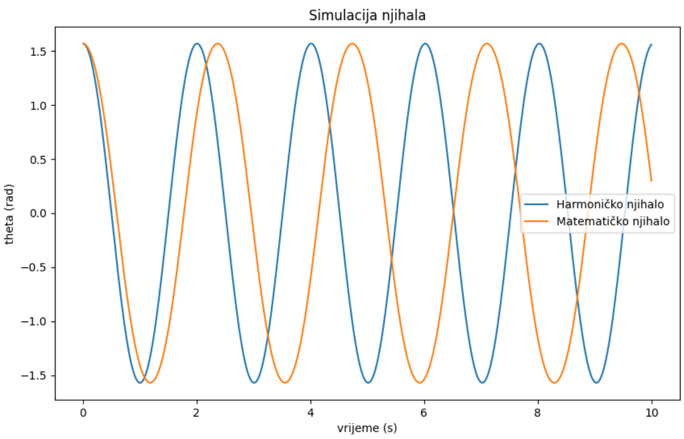
Radi lakše manipulacije velike količine podataka (30ak tisuća različitih vremena i otklona) napisali smo python skriptu koja uz pomoć **numpy-a** i **matplotlib-a** podatke obrađuje na način da učita sva tri stupca iz datoteke te prikaže vremenski stupac na x osi, a thetu na osi y. Radi laške usporedbe oba njihala su prikazana na istom grafu, različitim bojama.

Detaljan pregled koda je na mom githubu. `https://github.com/frantastic7/fortran`

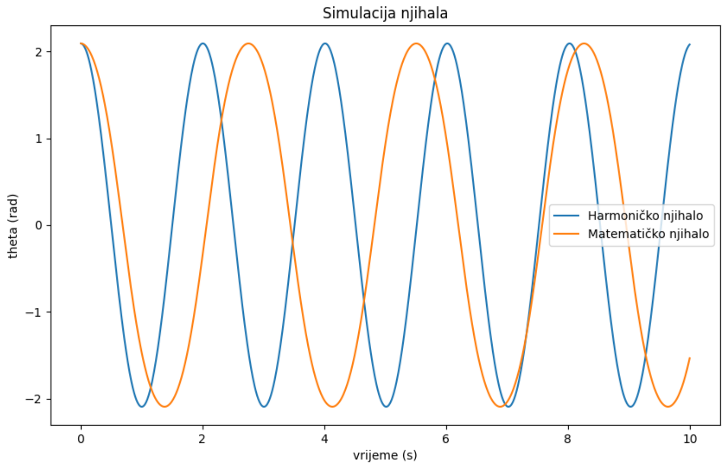
6.1 Grafovi simulacije



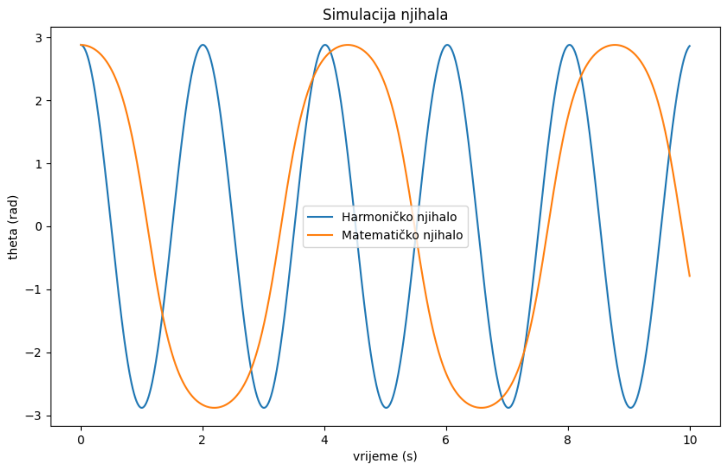
Slika 3: Simulacija njihala uz početni otklon od 45 stupnjeva



Slika 4: Simulacija njihala uz početni otklon od 90 stupnjeva



Slika 5: Simulacija njihala uz početni otklon od 120 stupnjeva



Slika 6: Simulacija njihala uz početni otklon od 165 stupnjeva

7 Zaključak

U ovom smo radu pokazali koliko aproksimacija malih kuteva utječe na pomak idealnog njihala naspram njegove realne varijante. Fortran je bio posebno prikladan za ovaj projekt pošto je simulacija, iako opsežna bila izvedena brzo s velikom float preciznošću.

Možemo primjetiti da su razlike u njihalima zaista velike, te da nam aproksimacije stvarno pomažu samo kada smo blizu lokalnog minimuma tj. točke ravnožete.

Iako je Fortran odličan za brzo i precizno računanje velikog broja podataka, za njihovu obradu potrebni su nam moderniji jezici i biblioteke poput pythona i numpya s kojima je rukovanje s velikim brojem podataka jednostavno i učinkovito.