1 DN3 - Matematično nihalo; Matija Ojo; 63200205

1.1 Uvod

Kotni odmik $\theta(t)$ (v radianih) pri nedušenem nihanju nitnega nihala opišemo z diferencialno enačbo:

$$\frac{g}{l}\sin(\theta(t)) + \theta''(t) = 0, \theta(0) = \theta_0, \theta'(0) = \theta_0'$$

kjer je $g = 9.80665 m/s^2$ težni pospešek, in l dolžina nihala.

1.2 Metode

Naloge se lotimo tako da enačbo drugega reda prevedemo na sistem prvega reda:

$$\theta'(t) = y(t)$$
$$y'(t) = -\frac{g}{l}\sin(\theta(t))$$

in rešujemo z metodo Runge-Kutta četrtega reda.

Runge-Kutta metode so družina implicitnih in eksplicitnih iterativnih metod, ki vključujejo Eulerjevo metodo, ki se uporablja v časovni diskretizaciji za približne rešitve simultanih nelinearnih enačb.

Najbolj uveljavljena je Runge-Kutta metoda 4. reda, ki rešuje aproksimira rešitev diferencialne enačbe.

Pri diferencialni enačbi želimo aproksimirati neznano funkcijo y, odvisna od časa t. Podano imamo začetno vrednost y_0 v začetnem času t_0 in funkcijo odvoda $f(t, y)$, velja torej:

$$\frac{dy}{dy} = f(t, y); \quad y(t_0) = y_0$$

Če želimo izračunati vrednost funkcije y v času t_1, to naredimo tako, da časovni interval diskretiziramo na n podintervalov dolžine h in uporabimo Runge-Kutta metodo 4. reda:

$$k_1 = hf(t, y)$$

$$k_2 = hf(t + \frac{h}{2}, y + \frac{k_1}{2})$$

$$k_3 = hf(t + \frac{h}{2}, y + \frac{k_2}{2})$$

$$k_4 = hf(t + h, y + k_3)$$

$$y_{n+1} = y_n + \frac{k_1 + 2k_2 + 2k_3 + k_4}{6}$$

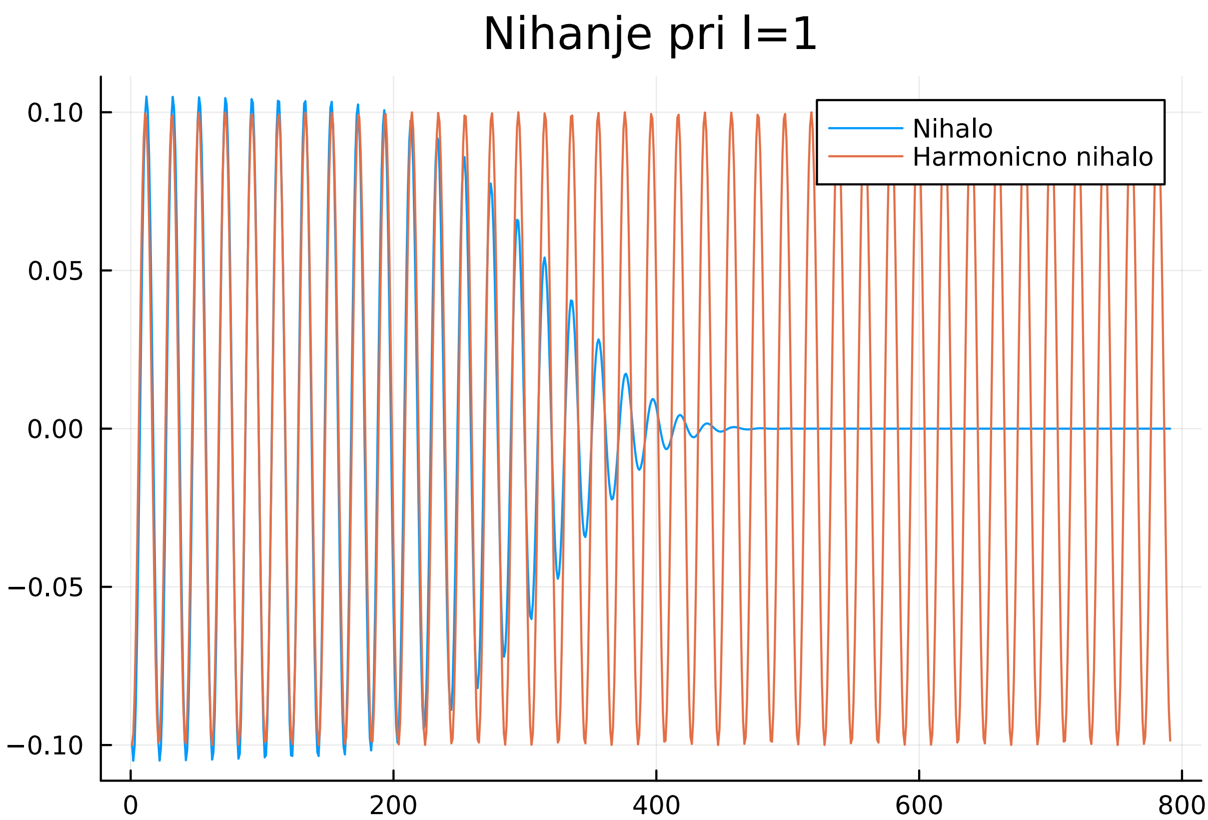
Poleg matematičnega nihala smo implementirali tudi harmonično nihalo, kjer se odmik v koraku $i+1$ izračuna kot:

$$\theta_{i+1} = \theta_0 \cos(i\sqrt{\frac{g}{l}})$$

1.3 Rezultati

Najprej primerjajmo rešitev z nihanjem harmonicnega nihala.

```
using DN3, Plots

n = 100
ts = 1:0.1:80

theta0 = 0.1
dtheta0 = 0.1

plot([nihalo(1, t, theta0, dtheta0, n) for t in ts], label="Nihalo")
plot!([nihalo_harmonicno(1, t, theta0, n) for t in ts], title="Nihanje pri l=1",
label="Harmonicno nihalo")
```

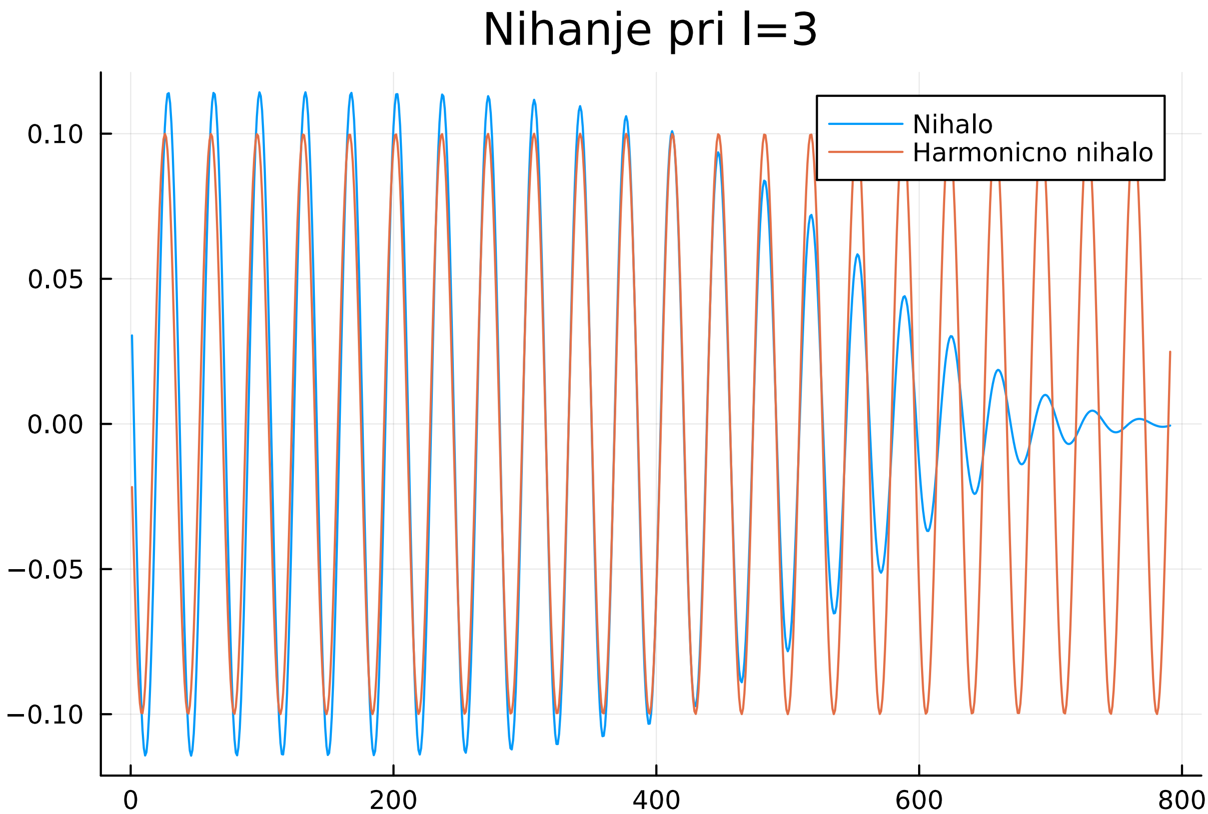



```
print()
```

Pri daljšem nihalu se frekvenca nihanja zmanjša, in se čas nihanja poveča.

```
plot([nihalo(3, t, theta0, dtheta0, n) for t in ts], label="Nihalo")
```

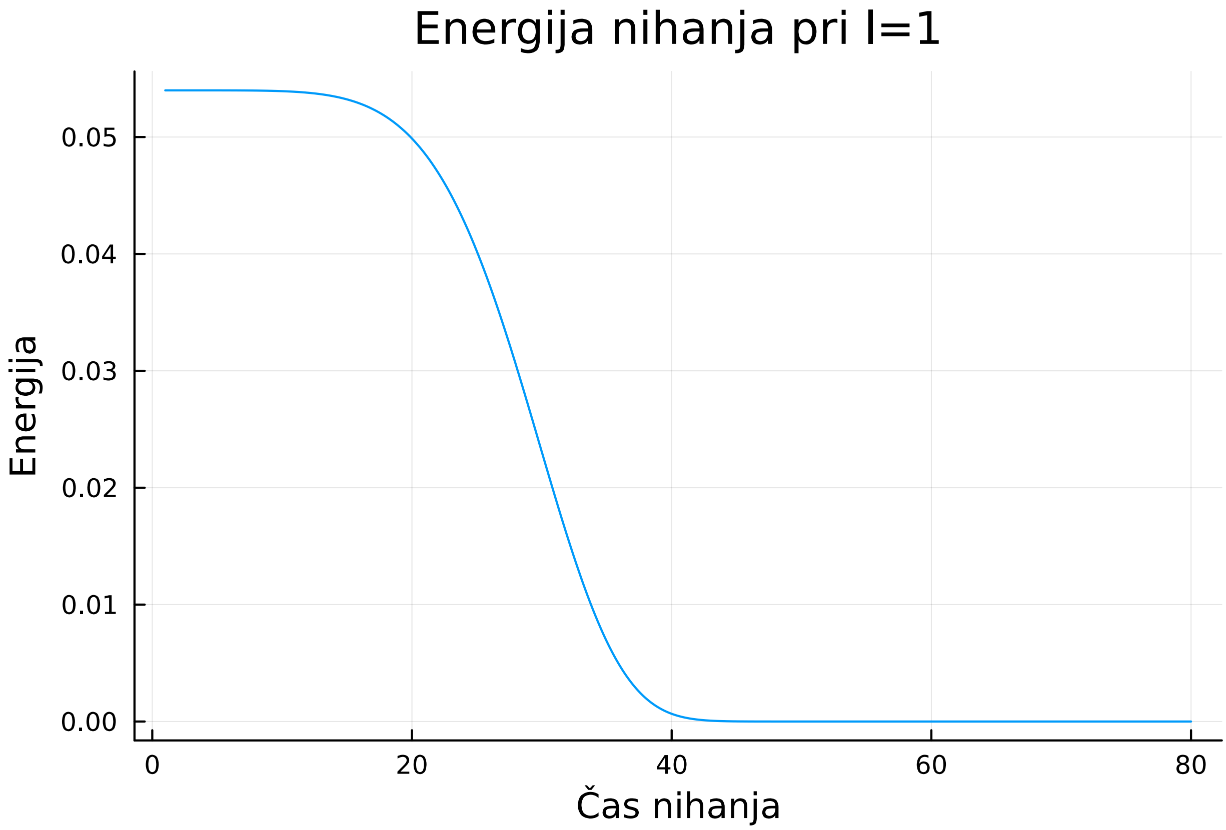
```
plot!([nihalo_harmonicno(3, t, theta0, n) for t in ts], title="Nihanje pri l=3",
label="Harmonicno nihalo")
```



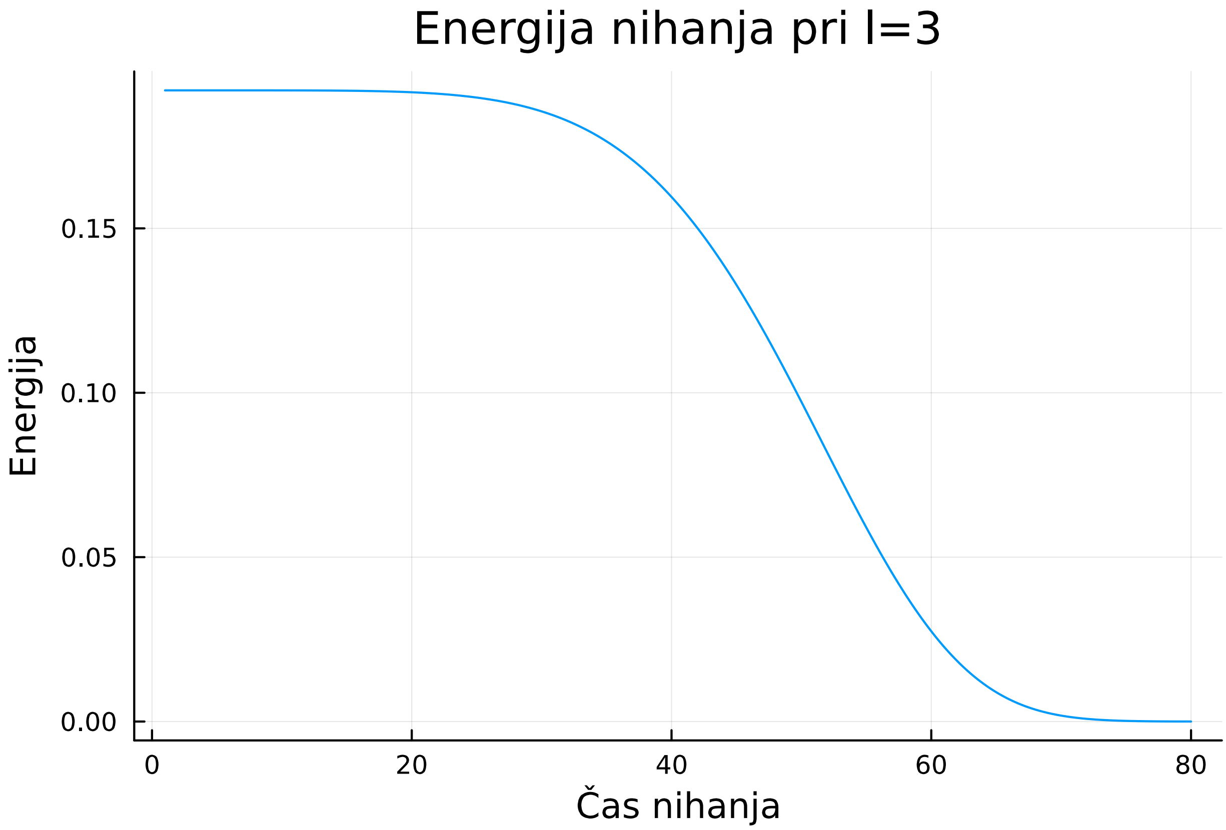
Kot vidimo iz zgornjih grafov, harmonicno nihanje ni odvisno od energije nihala, medtem ko je pri matematičnem nihalu to upoštevano. Posledica tega je, da bo matematično nihalo čedalje počasneje nihalo, saj se bo energija izgubljala.

Narišimo še graf, ki prikazuje kako se nihajni čas spreminja z energijo nihanja.

```
function energija(theta, dtheta, l)
  G = 9.80665
  m = 1
  kinetic = 0.5 * m * (l * dtheta)^2
  potential = m * G * l * (1 - cos(theta))
  return kinetic + potential
end

l = 1

energije = [
  let (theta, dtheta) = nihalo2(l, t, theta0, dtheta0, n)
    energija(theta, dtheta, l)
  end for t in ts
]
plot(ts, energije,
  title="Energija nihanja pri l=1", xlabel="Čas nihanja", ylabel="Energija",
legend=:none)
```




```julia
l = 3

energije = [
  let (theta, dtheta) = nihalo2(l, t, theta0, dtheta0, n)
    energija(theta, dtheta, l)
  end for t in ts
]
plot(ts, energije,
  title="Energija nihanja pri l=3", xlabel="Čas nihanja", ylabel="Energija",
legend=:none)
```



Kot vidimo iz zgornjih grafov, se energija nihanja zmanjšuje s časom. Pri večji dolžini nihala energija nihanja zmanjšuje ustrezno počasneje.