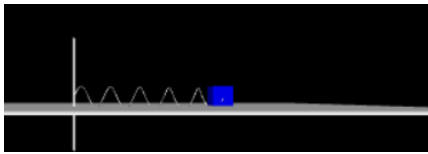


Harmonijski oscilator s prigušenjem

Jedan od učestalih problema koji se pojavljuju u mehanici je prigušeni harmonijski oscilator. Prigušeni harmonijski oscilator opisuje sustav koji periodično titra gdje se uz povratnu silu (najčešće elastičnu) pojavljuje i određena sila otpora poput sile trenja, otpora zraka, fluida i sl. Za razliku od idealnog oscilatora koji bi titrao beskonačno, prigušeni oscilator titrat će neko vrijeme te će, kada sustav preda svu energiju okolini, prestati titrati. Razlikuju se tri različita oblika prigušenja koje se može pojaviti – podkritično prigušenje, kritično prigušenje i nadkritično prigušenje. Oscilatoru koji ima podkritično prigušenje postepeno će se smanjivati amplituda titranja. Što je prigušenje jače, to će se amplituda titranja više smanjivati i naposljetku postati nula. Harmonijski oscilator s kritičnim prigušenjem vratit će se u ravnotežni položaj te više neće titrati, a oscilator s nadkritičnim prigušenjem ponašat će se vrlo slično oscilatoru s kritičnim prigušenjem, međutim tijelo će se sporije vratiti u početni položaj.



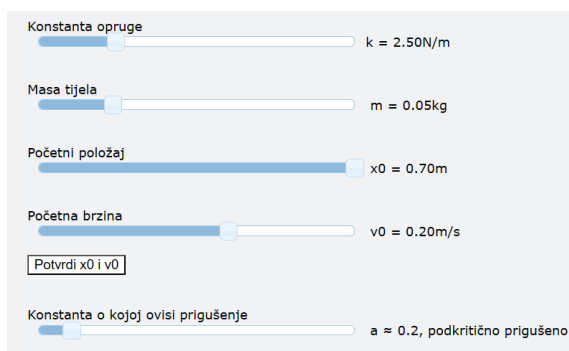
Za prikaz modela harmonijskog oscilatora s prigušenjem izabran je sustav s oprugom i tijelom koje horizontalno titra na podlozi.

Ova simulacija realizirana je u programu Python, a korištena je i nadogradnja VPython kojom je omogućen prikaz samog grafičkog sučelja, crtanje grafova i mijenjanje iznosa parametara.

```
delta_t=0.0005
t=0
k=0.5
Lo=1.5
m=0.02
b=0
v0 = 0
a=0 #prvo stavljam oscilator bez prigušenja
b=0
beta=0
w0=0
```

Na početku koda postavljeni su početni uvjeti koji omogućuju da simulacija započne pri samome pokretanju programa. Definirane su vrijednosti parametara pomoću kojih se regulira ponašanje modela, određene su vrijednosti za masu ($m=0.02\text{kg}$), konstantu opruge ($k=0.5\text{N/m}$), faktor prigušenja ($\beta=0$), početnu brzinu ($v_0=0$) i početni položaj ($x_0=-0.65$) te duljinu opruge koja će se mijenjati ovisno o položaju tijela.

Također, određena je vrijednost `delta_t` koja kasnije u `while` petlji omogućava odvijanje simulacije tj. omogućava da se račun provodi za svaku razliku vremena iznosa `delta_t`. Kako bi vrijednosti koje se računaju bile što točnije, a grafovi koji se iscrtavaju što precizniji uzeta je vrlo malena vrijednost za `delta_t`.



Korisniku je omogućena promjena početnih uvjeta tijekom odvijanja simulacije. Interaktivnim sučeljem omogućava se lakše uviđanje ovisnosti parametara koji se iscrtavaju u grafovima (energija, položaj, brzina, akceleracija) o parametrima koji se mogu mijenjati tijekom simulacije (masa, konstanta opruge, početni položaj, početna brzina, koeficijent prigušenja).

```
w0 = sqrt(k / m)
beta = a * w0 #ovaj a mi govori kakvo je prigušenje
b = 2 * m * beta
```

S obzirom na to da je korisniku omogućena i promjena prigušenja, važno je sprovesti račun za

to prigušenje. Naime, vrsta prigušenja se određuje tako da ako je beta koja označava koeficijent prigušenja:

- 1) jednaka nuli, nema prigušenja
- 2) manja od konstante w_0 , prigušenje je podkritično
- 3) jednaka konstanti w_0 , prigušenje je kritično
- 4) veća od konstante w_0 , prigušenje je nadkritično

U kasnijoj računici upotrebljavat će se b koji iznosi $2 \times m \times \beta$ radi jednostavnijeg zapisa.

```
Ke=0.5/m*mag(ptijela)**2
Pe=0.5*k*x**2
Emech=Pe+Ke
Keblock.plot( pos=(t,Ke) )
Peblock.plot( pos=(t,Pe) )
Emechblock.plot(pos=(t,Emech))

Fopruga=-k*x*Lhat
Fotpora=b*(-ptijela/m)
ptijela=ptijela+(Fopruga+Fotpora)*delta_t
tijelo.pos=tijelo.pos+ptijela*delta_t/m
opruga.axis=tijelo.pos-zid.pos
L=tijelo.pos
Lhat=hat(L)
x=mag(L)-Lo

#za pomak, brzinu i akceleraciju graf
akc = -b * v / m - k * x / m #prema formuli
v = v + akc * delta_t
```

Glavni račun kojim se provodi promjena parametara tijekom simulacije i time omogućava dosljedno crtanje grafova smješteno je u while petlji koja se odvija dok god je program pokrenut.

Svaki Δt vremena iznova se računa trenutni iznos kinetičke energije, potencijalne energije, mehaničke energije, položaja tijela te iznos njegove brzine i akceleracije.

Trenutna kinetička energija računa se pomoću formule $\frac{mv^2}{2}$, gdje je $\text{mag}(\text{ptijela})$ iznos vektora brzine. Trenutna potencijalna energija računa formulom $\frac{kx^2}{2}$.

Ukupna energija tijela, mehanička energija, jednaka je zbroju kinetičke i potencijalne. Zatim, računaju se i sile uz pomoć kojih se reguliraju vrijednosti koje će utjecati na prikaz tijela u simulaciji (elastična sile, sila otpora) te se zato moraju koristiti vektorske vrijednosti, točnije potrebno je paziti na orijentaciju vektorskih veličina koje se mijenjaju tijekom odvijanja simulacije. Dakle, provodi se račun za trenutni položaj tijela. Uz to, računaju se i brzina i akceleracija pomoću formula:

$$a = \frac{-bv}{m} - \frac{kx}{m}$$

$$v = v_{\text{prijašnja}} + a\Delta t$$

U simulaciji je omogućeno pratiti promjenu ukupne energije sustava, ali i promjene kinetičke i potencijalne energije, promjenu položaja tijela, brzine i akceleracije jer se iscrtavaju u grafovima u sučelju. Naime, promjenom početnih uvjeta (konstante opruge, mase tijela, početnog položaja, početne brzine i vrste prigušenja) korisnik može uočiti ovisnost energije, brzine sustava, trenutnog položaja i akceleracije o izabranim parametrima. Stoga, simulacija je namijenjena uočavanju navedene povezanosti te pomoću interaktivnog sučelja nadogradnji znanja o ponašanju harmonijskog oscilatora koje je ovisno o različitim parametrima. Simulacijom se također nastoji što dosljednije prikazati sam model harmonijskog oscilatora s različitim vrstama prigušenja.

Marija Šporčić