

Proučavanje Newtonovih jednadžbi gibanja pomoću animacija u programskom jeziku Python u osnovnom i srednjoškolskom obrazovanju

Ines Vlahović¹, Matko Glunčić¹, Vladimir Paar^{1,2}

¹ *Fizički odsjek, Prirodoslovno-matematički fakultet, Sveučilište u Zagrebu*

² *Hrvatska akademija znanosti i umjetnosti*

Sažetak. Zadnjih godina programski jezik Python (<https://www.python.org>) je postao općeprihvaćen od velikog broja znanstvenika u svijetu. Njegovo korištenje ne zahtijeva kupnju licenci koje predstavljaju dodatne troškove školama. Prednost su i razne dodatne ugrađene biblioteke kao što su NumPy i SciPy (<https://en.wikipedia.org/wiki/NumPy>) te „Matplotlib“ (http://matplotlib.org/api/animation_api.html). Upravo zbog raširenosti ovog programskog jezika među znanstvenicima, njegovo uvođenje kao dodatnog alata za primjenu u nastavi fizike čini se kao razumno rješenje (pogotovo kada se tijekom razvoja novog kurikula govori i o uvođenju informatike kao obaveznog predmeta u osnovne škole). U ovom radu prikazana je njegova primjena s ciljem razvijanja učeničkog razumijevanja i sposobnosti računalnog rješavanja Newtonovih jednadžbi gibanja, pomoću Eulerove metode za zadane početne uvjete. Dajemo prijedlog za primjenu u školama u okviru raznih nastavnih cjelina predmeta Fizike, uz razvijanje dodatnih aplikacija (skripti u Pythonu) unutar jednog paketa aplikacija. Taj paket bi se mogao upotrijebiti za istraživački tip nastave, za razvijanje i usvajanje fizikalnih koncepata te ispravljanje miskoncepcija, u kombinaciji s realnim pokusima ili kao samostalni istraživački alat, osobito u slučajevima gdje su realni pokusi složeni ili teško ostvarivi. Osobito korisna strana takvog pristupa je mogućnost da se računalno vođeni pokusi izvode i samostalno kod kuće, a ne samo na nastavi. Osim ujednačenosti razvijanja kritičkog razmišljanja u školama, prednost razvoja spomenutog paketa bila bi i usklađenost poučavanja fizike (i programiranja) od osnovne i srednje škole pa do fakulteta. Naime, uporabom simulacija učenici bi mogli objašnjavati prirodne pojave poput znanstvenika. Kao konkretan primjer prikazujemo metodički oblikovan pristup obradi jednolikog gibanja po kružnici uz pomoć Pythona.

Ključne riječi: Python, aplikacije, Newtonove jednadžbe gibanja, istraživačka nastava.

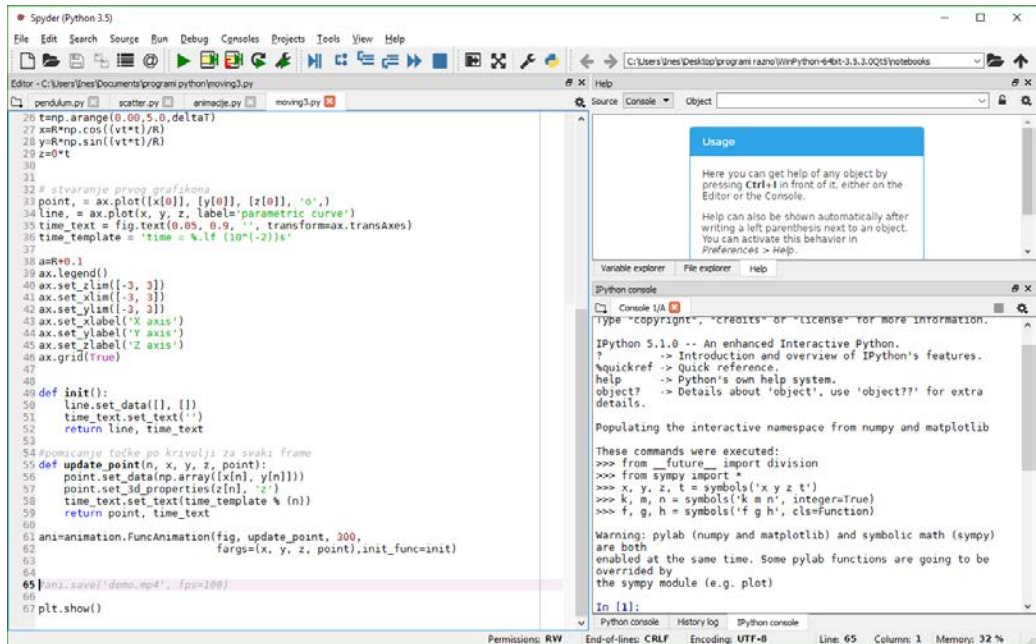
UVOD

U današnje vrijeme računalni pokusi i simulacije u nastavi postaju sve bitniji dodatan alat za proučavanje prirodnih zakona. Postoji brojna literatura, npr. [1], te web stranice s gotovim skriptama i aplikacijama, napisanim u različitim programskim jezicima, na kojima su prikazane simulacije pokusa iz različitih područja fizike, u različitim govornim jezicima. Kako bi učenicima omogućili što interaktivniji pristup istraživačkoj nastavi, koja bi se mogla odvijati i na satu fizike (uz pomoć profesora/profesorica) i samostalno kod kuće, predlažemo razvijanje univerzalnog paketa programa za osnovne i srednje škole u programskom jeziku Python. Prednosti Pythona su njegovo jednostavno korištenje te pristupačnost koja bi omogućila i jednostavnije i lakše stručno osposobljavanje profesora za rad. Dodatna prednost bi bilo povezivanje sadržaja fizike s nastavom predmeta informatike i matematike (pogotovo pošto se planira uvesti predmet informatika kao obavezan predmet u osnovnim školama).

Bitno pitanje je zadovoljavaju li simulacije (modela), kao dodatan alat, četiri bitna koraka u istraživačkoj nastavi – formulaciju problema, promatranje i eksperiment, interpretaciju podataka (postavljanje hipoteze) te dodatno testiranje i provjeru hipoteze (koja ako vrijedi,

postaje teorija). Mi ćemo na primjeru jednolikog kružnog gibanja pomoću programskog jezika Python prikazati te korake.

METODE



SLIKA 1. Spyder sučelje. U njemu se automatski vidi dio za unos koda (lijevi dio sučelja) i iPython konzola za pokretanje programa te prikaz izvršenja programa (desni dio sučelja).

U ovom radu koristi se programski jezik Python, verzija 3.5, te njegov interpreter Spyder iz WinPython paketa ([2], slika 1).

Kao predložak, koristili smo dio gotovog koda koji se može pronaći na web stranicama python modula specijaliziranih za izradu znanstvenih proračuna, animacija, crtanja grafikona, od kojih su neki korišteni u ovom primjeru (Matplotlib ([3], [4]), NumPy ([5], [6])).

Za računanje gibanja tijela u gravitacijskom polju, često se u simulacijama koristi Eulerova metoda koja na temelju zadanih početnih uvjeta i malog vremenskog intervala može odrediti poziciju, brzinu, akceleraciju tog tijela u slijedećem koraku kao što prikazuje tablica 1 za slučaj 1D gibanja (radi jednostavnosti). Analogno se može napraviti i tablica za 3D slučaj u kojoj bi, naravno, se trebali računati koraci za svaku komponentu koordinatnog sustava.

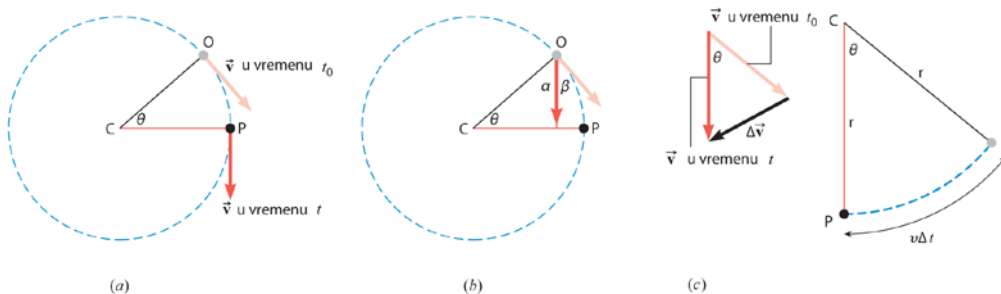
TABLICA 1. Prikaz postupka određivanja koraka pomoću Eulerove metode za 1D slučaj

Korak	Vrijeme	Položaj	Brzina	Akceleracija
0	t_0	x_0	v_0	$a_0 = F(x_0, v_0, t_0)/m$
1	$t_1 = t_0 + \Delta t$	$x_1 = x_0 + v_0 \cdot \Delta t$	$v_1 = v_0 + a_0 \cdot \Delta t$	$a_1 = F(x_1, v_1, t_1)/m$
2	$t_2 = t_1 + \Delta t$	$x_2 = x_1 + v_1 \cdot \Delta t$	$v_2 = v_1 + a_1 \cdot \Delta t$	$a_2 = F(x_2, v_2, t_2)/m$
:	:	:	:	:
n	$t_n = t_{n-1} + \Delta t$	$x_n = x_{n-1} + v_{n-1} \cdot \Delta t$	$v_n = v_{n-1} + a_{n-1} \cdot \Delta t$	$a_n = F(x_n, v_n, t_n)/m$

JEDNOLIKO KRUŽNO GIBANJE

Kao i za slučaj jednolikog gibanja po pravcu, tako i za slučaj jednolikog kružnog gibanja, vrijede Newtonovi zakoni tj. na to tijelo mora djelovati neka sila. Ta sila koja djeluje na tijelo da bi ono ostalo u kružnoj putanji pri konstantnoj brzini je centripetalna sila koja je uvijek usmjerena prema središtu rotacije tog tijela i okomita na smjer brzine tj. gibanja. Kako bi proučavali preko drugog Newtonovog zakona kako polumjer kruženja ovisi o sili, masi i akceleraciji u kružnom gibanju, trebamo doći do izraza za akceleraciju – centripetalnu akceleraciju. Jednostavan način za izvođenje izraza za centripetalnu akceleraciju - $a_{cp}=v^2/r$ se može napraviti preko sličnosti trokuta tj. jednakosti (slika 2):

$$\frac{\Delta v}{v} = \frac{v \cdot \Delta t}{r} \quad (1)$$



SLIKA 2. a) Prikaz vektora brzine u trenutku t i t_0 . b) translacija vektora brzine iz točke P u O daje nam uvid da se ta dva vektora brzine razlikuju za c) prijađeni kut θ . Slika je preuzeta i prerađena s [7].

Iz tog proizlazi izraz za centripetalnu silu, preko drugog Newtonovog zakona:

$$F = m \cdot a = m \cdot \frac{v^2}{r} \quad (2)$$

Simulacija kružnog gibanja u programskom jeziku Python

Ova simulacija kružnog gibanja računa pozicije u 3D prostoru (premda se proučava 2D gibanje, stoga se z komponenta može postaviti na 0 ili neku drugu konstantnu vrijednost), preko ulaznih parametara za centripetalnu silu, masu i tangencijalnu brzinu. Iz tih vrijednosti prvo se računa polumjer kruženja r za izračunavanje koordinata x i y preko relacija:

$$x = r \cdot \cos(\theta), y = r \cdot \sin(\theta), \quad (3)$$

pri čemu je:

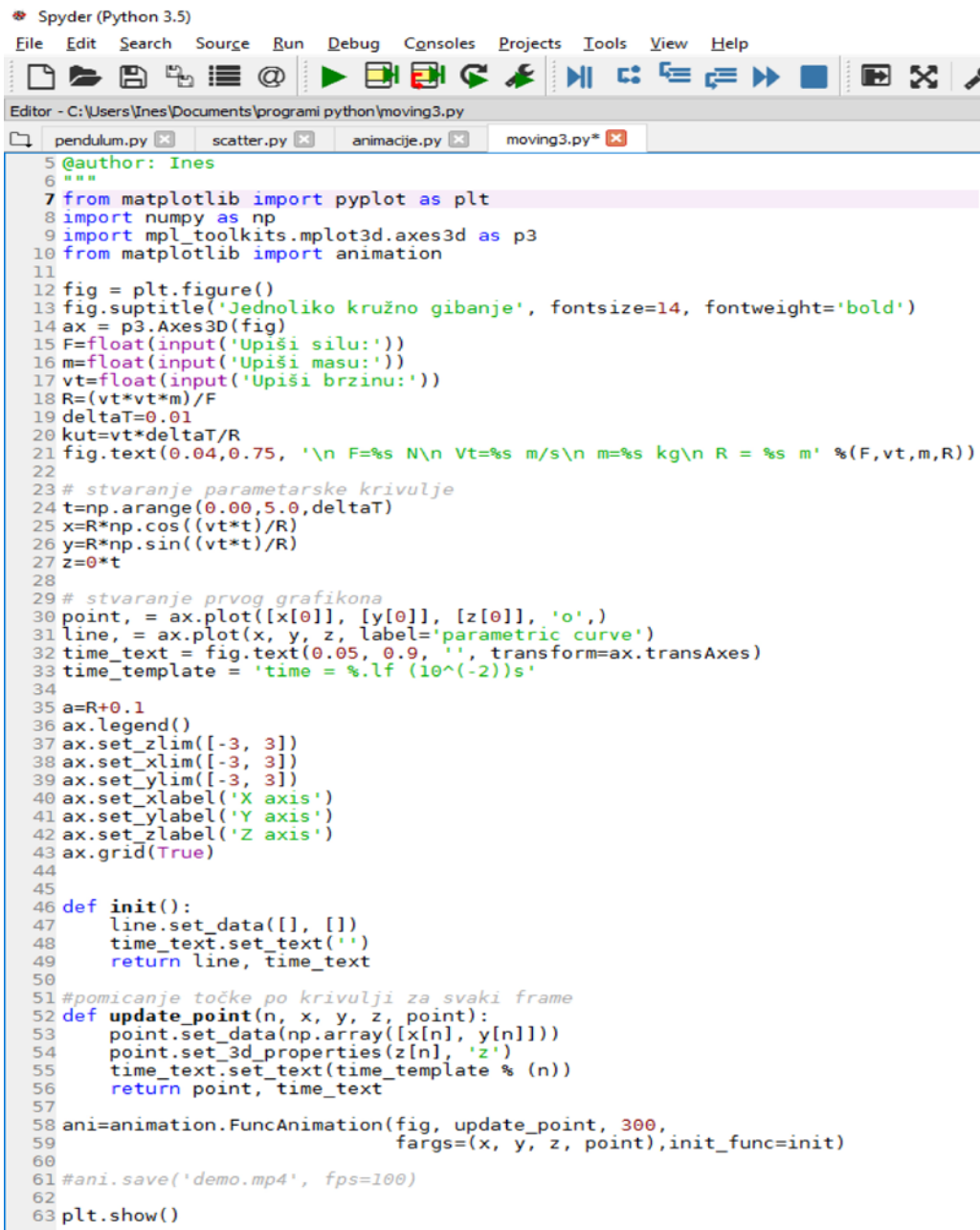
$$\theta = (v \cdot \Delta t) / r, \quad (4)$$

a Δt kratki vremenski interval koji je u simulaciji postavljen na 0.01s, i može se promijeniti u kodu. Kod simulacije je prikazan na slici 3.

Simulacije smo izvodili za tri slučaja, kako bi se došlo do zaključaka o ovisnosti polumjera kruženja tijela o masi, sili i brzini:

- 1.) za konstantnu silu i brzinu,
- 2.) za konstantnu brzinu i masu,
- 3.) za konstantnu silu i masu.

Za svaki od ovih slučajeva predložimo izvođenje po najmanje tri do pet simulacija, pri kojima se može na interaktivan način s učenicima odrediti polumjer preko 3D ili 2D prostora u simulaciji (premda je i sam izračunat i u ovoj verziji, prikazan).



```

5 @author: Ines
6 """
7 from matplotlib import pyplot as plt
8 import numpy as np
9 import mpl_toolkits.mplot3d.axes3d as p3
10 from matplotlib import animation
11
12 fig = plt.figure()
13 fig.suptitle('Jednoliko kružno gibanje', fontsize=14, fontweight='bold')
14 ax = p3.Axes3D(fig)
15 F=float(input('Upiši silu:'))
16 m=float(input('Upiši masu:'))
17 vt=float(input('Upiši brzinu:'))
18 R=(vt*vt*m)/F
19 deltaT=0.01
20 kut=vt*deltaT/R
21 fig.text(0.04,0.75, '\n F=%s N\n Vt=%s m/s\n m=%s kg\n R = %s m' %(F,vt,m,R))
22
23 # stvaranje parametarske krivulje
24 t=np.arange(0.00,5.0,deltaT)
25 x=R*np.cos((vt*t)/R)
26 y=R*np.sin((vt*t)/R)
27 z=0*t
28
29 # stvaranje prvog grafikona
30 point, = ax.plot([x[0]], [y[0]], [z[0]], 'o',)
31 line, = ax.plot(x, y, z, label='parametric curve')
32 time_text = fig.text(0.05, 0.9, '', transform=ax.transAxes)
33 time_template = 'time = %.1f (10-2)s'
34
35 a=R+0.1
36 ax.legend()
37 ax.set_zlim([-3, 3])
38 ax.set_xlim([-3, 3])
39 ax.set_ylim([-3, 3])
40 ax.set_xlabel('X axis')
41 ax.set_ylabel('Y axis')
42 ax.set_zlabel('Z axis')
43 ax.grid(True)
44
45
46 def init():
47     line.set_data([], [])
48     time_text.set_text('')
49     return line, time_text
50
51 #pomicanje točke po krivulji za svaki frame
52 def update_point(n, x, y, z, point):
53     point.set_data(np.array([x[n], y[n]]))
54     point.set_3d_properties(z[n], 'z')
55     time_text.set_text(time_template % (n))
56     return point, time_text
57
58 ani=animation.FuncAnimation(fig, update_point, 300,
59                             fargs=(x, y, z, point),init_func=init)
60
61 #ani.save('demo.mp4', fps=100)
62
63 plt.show()

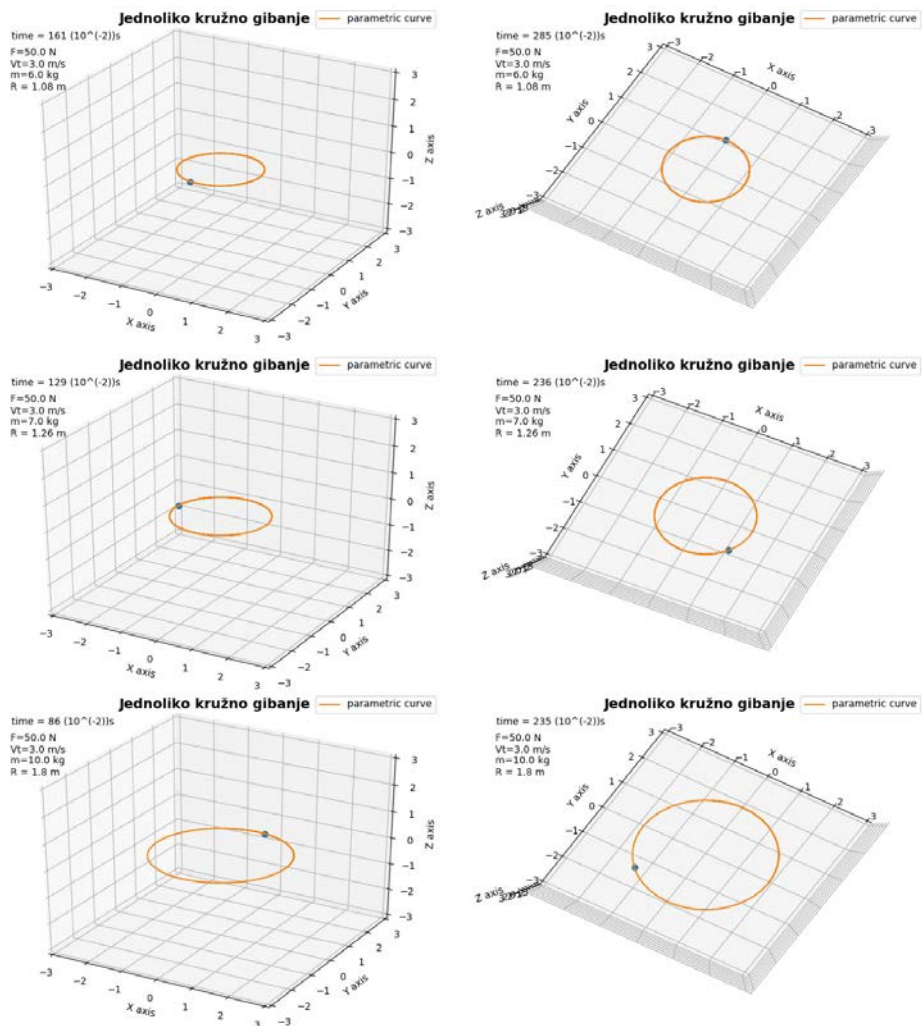
```

SLIKA 3. Kod simulacije.

IZVOĐENJE SIMULACIJA NA NASTAVI – FORMULACIJA PROBLEMA I EKSPERIMENT

Kako ovisi polumjer u jednolikom kružnom gibanju o masi?

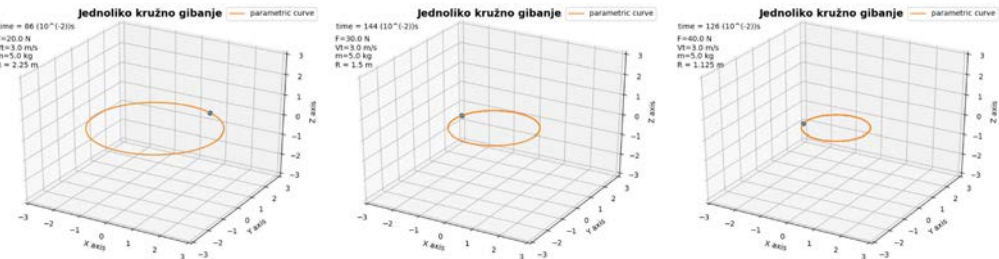
Kao prvi primjer prikazat ćemo u našoj simulaciji ovisnosti polumjera jednolikog kružnog gibanja o masi. Same simulacije mogu se izvoditi ili uz pomoć profesora (gdje profesor pokreće simulacije, a učenici očitavaju promjene u polumjeru ovisno o zadanim parametrima gibanja) ili su u mogućnosti samostalno izvoditi računalne simulacije (u školi ili kod kuće) za potrebe dodatnog istraživanja i razvoja kritičkog razmišljanja. Kao primjer možemo odabrati za ulazne parametre silu od 50 N i brzinu od 3 m/s. Pokretanjem simulacije za masu ćemo odabrati 6 kg, 7 kg i 10 kg. Učenici će moći direktno sa simulacije mjeriti polumjer (premda je na samoj simulaciji i napisan u ovoj verziji), zbog mogućnosti rotacije u 3D prostoru u Pythonu. Rezultati takvih simulacija su prikazani na slici 4.



SLIKA 4. Prikaz promjene polumjera u ovisnosti o masi, uz konstantnu brzinu i silu. U lijevom stupcu slike nalaze se simulacije u 3D-u, a u desnom stupcu njihove projekcije u 2D odnosno u ravninu x-y iz koje se mogu mjeriti promjeri odnosno polumjeri.

Kako polumjer kruženja u jednolikom kružnom gibanju ovisi o primijenjenoj sili?

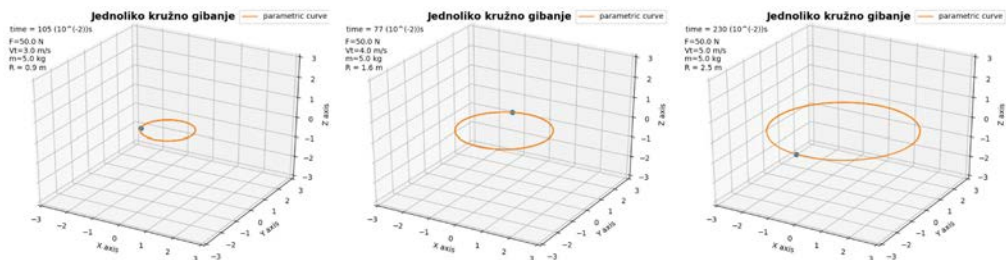
Kao drugi primjer proučavanja kružnog gibanja, mogla bi se proučavati ovisnost primijenjene sile na tijelo u jednolikom kružnom gibanju, uz konstantnu zadanu brzinu i masu. Učenici bi prilikom svake simulacije mogli bilježiti očitani polumjer, kao i u prethodnom primjeru. Npr. možemo odabrati za ulazne parametre za masu tijela 5 kg i njegovu tangencijalnu brzinu 3 m/s, a vrijednosti sile za svaku simulaciju 20, 30 i 40 N (slika 5).



SLIKA 5. Prikaz promjene polumjera u ovisnosti o sili, uz konstantnu brzinu i masu.

Kako polumjer kruženja u jednolikom kružnom gibanju ovisi o tangencijalnoj brzini?

Kako bi učenici mogli doći do ovog odgovora, pomoću simulacije za ovaj slučaj možemo postaviti kao ulazne parametre iznose za centripetalnu silu od 50 N i masu od 5 kg. Zatim izvodimo tri simulacije za tangencijalne brzine v_1 - v_3 , redom od 3-5 m/s. Prilikom svake simulacije učenici mogu uz pomoć 2D prikaza u 3D prostoru preko oznaka za osi očitati polumjer te ga zapisati. Rezultati za ovaj slučaj prikazani su na slici 6 (osim slika unutar programa može se snimiti i video za kasniju upotrebu, tako da se obriše oznaka # - komentara u kodu ispred reda 61 (slika 3), ali u tom slučaju za izvršenje simulacije potrebno je malo dulje vrijeme).



SLIKA 6. Prikaz promjene polumjera u ovisnosti o brzini, uz konstantnu silu i masu.

INTERPRETACIJA PODATAKA

Nakon formulacije problema koji želimo proučavati kod jednolikog kružnog gibanja, te provođenja interaktivne nastave prilikom koje su učenici mogli mjeriti polumjere sa simulacija očitavanjem podataka sa simulacije u 3D ili 2D prostoru (koje se dodatno može također izračunati i preko koordinata sa x, y, z osi prikazanih pri postavljanju kursora na pojedini dio putanje – povezivanje sa sadržajima nastave matematike), omogućeno je dolaženje do

zaključaka. Iz podataka dobivenih pomoću simulacija, preko prikaza u 3D prostoru (slika 4) učenici lako mogu doći do zaključka da je polumjer kruženja proporcionalan masi, odnosno da vrijedi relacija $r \sim m$. Također, iz slike 5, učenici mogu zaključiti da polumjer ovisi obrnuto proporcionalno o sili tj. $r \sim 1/F$. Iz slike 6, učenici mogu zaključiti da polumjer raste sa brzinom, a iz očitanih podataka za polumjer mogu doći do zaključaka da je ta ovisnost $r \sim v^2$. Iz tih ovisnosti sami bi mogli doći do izraza za polumjer u kružnom gibanju (do na konstantu):

$$r = \frac{v^2 \cdot m}{F} \quad (5)$$

TESTIRANJE I PROVJERA HIPOTEZE

Nakon interpretacije podataka, odnosno postavljanja hipoteze relevantno za kružno gibanje, učenici bi mogli na nastavi ili samostalno provjeravati relaciju 5, na nizu zadanih vrijednosti za sva tri proučavana slučaja radi dodatne potvrde.

ZAKLJUČAK

Na ovom primjeru vidimo da je nastavu fizike moguće izvoditi pomoću računalnih alata tj. simulacija. Jednostavno prikazane i izvedive simulacije, koje su pristupačne i profesorima i učenicima, mogu pomoći u razumijevanju prirodnih zakonitosti preko istraživačke nastave. Ovaj primjer jednolikog kružnog gibanja zadovoljava kriterije bitne za takav tip izvođenja nastave preko formulacije problema, njegovog proučavanja i postavljanja (kao u realno izvođenim pokusima) preko kojih se može doći do interpretacije tako dobivenih podataka vezanih uz određene prirodne zakone. Razvojem cjelovitog paketa takvih simulacija kao dodatnog alata za različita područja fizike uvelike bi podiglo razinu razumijevanja fizikalnih pojava, a pogotovo onih područja gdje je izvođenje realnih pokusa otežano ili onemogućeno. Bitnu ulogu i doprinos u razvoju takvih paketa računalnih alata mogao bi biti omogućen preko diplomskih radova na fakultetima te dogovora o temama pogodnim za istraživačku nastavu preko simulacija sa profesorima u osnovnim i srednjim školama (odnosno prijedloga cjelovitog kurikula predmeta fizike, kemije, informatike i matematike) u kojima bi se takva nastava i provodila.

ZAHVALA

Zahvaljujemo se doc.dr.sc. Ivici Martinjaku na savjetima.

LITERATURA

1. Vladimir Paar, Fizika 1, udžbenik za prvi razred gimnazije, Školska knjiga, Zagreb
2. <https://winpython.github.io>
3. J. D. Hunter, "Matplotlib: A 2D Graphics Environment," in Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007.
4. <http://matplotlib.org>
5. <http://www.numpy.org>
6. <http://stackoverflow.com/questions/31303601/animate-a-python-yplot-by-moving-a-point-plotted-via-scatter>
7. J. D. Cutnell; K.W. Johnson, Physics, Wiley, 2006.