Projektni zadatak: Metode matematičke fizike

Tekst zadatka:

Lopta mase m se izbacuje početnom brzinom v_0 pod kutem θ prema horizontali (kosi hitac). Giba se pod utjecajem sile teže i otpora zraka iznosa $m\alpha v$ ($\alpha > 0$).

a) Odredite gibanje lopte (usmjerite os Oy vertikalno prema gore). Pokažite da se prelaskom na limes $\alpha \to 0$ dolazi do gibanja bez otpora zraka (klasični kosi hitac).

b) Označimo s A najvišu točku putanje lopte. Ako je $\alpha = g/v_0$ izračunajte kut θ uz koji je x-koordinata točke A maksimalna.

c) Neka je $g = 9.81$, $v_0 = 20$ i $\theta = \frac{\pi}{4}$. Numerički odredite najvišu točku putanje lopte uz $\alpha = 0.5$ – označimo tu točku s B. Ako je $\alpha = 0.2$ numerički odredite kut θ uz koji lopta pogađa točku B (odredite sva rješenja).

Rješenje

Napomene.

- Gibanje lopte aproksimiramo gibanjem točke jednake mase, tj. zanemarujemo utjecaj rotacije lopte na gibanje.
- Gibanje je ravninsko

a) U ovom zadatku promatramo dvodimenzionalno gibanje. Radijvektor položaja lopte je

$$\vec{r}(t) = (x(t), y(t), 0) = x(t)\vec{i} + y(t)\vec{j}$$

Sile koje djeluju na tijelo su:

- Sila teže: $\vec{G} = -mg\vec{j}$
- Sila otpora zraka: $\vec{F}_o = -m\alpha v = -m\alpha\dot{x}\vec{i} - m\alpha\dot{y}\vec{j}$

Iz drugog Newtonovog aksioma slijedi:

$$m\vec{a} = \vec{G} + \vec{F}_o$$

$$m\ddot{\vec{r}}(t) = -mg\vec{j} - m\alpha\dot{x}(t)\vec{i} - m\alpha\dot{y}(t)\vec{j}$$

Raspisom po komponentama dobivamo:

$$\vec{i} \ldots m\ddot{x} = -m\alpha\dot{x}$$

$$\vec{j} \ldots m\ddot{y} = -mg - m\alpha\dot{y}$$

$$\vec{i} \ldots \ddot{x} = -\alpha\dot{x}$$

$$\vec{j} \ldots \ddot{y} = -g - \alpha\dot{y}$$

uz početni položaj i brzinu

$$\vec{r}(0) = (x(0), y(0), 0) = (0, 0, 0)$$

$$\vec{v}(0) = (\dot{x}(0), \dot{y}(0), 0) = (v_0 cos\theta, v_0 sin\theta, 0)$$

gdje je $v_0 = |\vec{v_0}|$

Rješavamo:

$$\begin{cases} \ddot{x} = -\alpha\dot{x} \\ x(0) = 0 \\ \dot{x}(0) = v_0 \cos\theta \end{cases}$$

$$\begin{cases} \ddot{y} = -\alpha\dot{y} - g \\ y(0) = 0 \\ \dot{y}(0) = v_0 \sin\theta \end{cases}$$

Rješavanjem sustava dobivamo

$$x(t) = \frac{v_0 cos\theta}{\alpha}(1 - e^{-\alpha t})$$

$$\dot{x}(t) = v_0 cos\theta \cdot e^{-\alpha t}$$

$$y(t) = (\frac{v_0 sin\theta}{\alpha} + \frac{g}{\alpha^2})(1 - e^{-\alpha t}) - \frac{g}{\alpha}t$$

$$\dot{y}(t) = (v_0 sin\theta + \frac{g}{\alpha})e^{-\alpha t} - \frac{g}{\alpha}$$

Prelazak na limes

Ako α teži prema 0, otpor zraka možemo zanemariti te se u tom slučaju jednadžbe se svode na:

$\ddot{x} = 0$

$\ddot{y} = -g$

Integriranjem slijedi:

$$x(t) = v_0 \cos\theta \cdot t$$

$$\dot{x}(t) = v_0 \cos\theta \cdot$$

$$y(t) = v_0 \sin\theta \cdot t - \frac{1}{2}gt^2$$

$$\dot{y}(t) = v_0 \sin\theta \cdot -gt$$

Time smo dobili jednadžbe za klasični kosi hitac

Analizirajmo rješenje početnog sustava kad $\alpha \to 0$

$$\lim_{\alpha\to 0} \frac{v_0 cos\theta}{\alpha}(1 - e^{-\alpha t}) = v_0 cos\theta \cdot t \lim_{\alpha\to 0} \frac{e^{-\alpha t} - 1}{-\alpha t}$$

Iz $\lim_{x\to 0} \frac{e^x - 1}{x} = 1$ slijedi

$$v_0 cos\theta \cdot t \lim_{\alpha\to 0} \frac{e^{-\alpha t} - 1}{-\alpha t} = v_0 cos\theta \cdot t$$

Za ispitivanje uniformne konvergencije možemo definirati niz funkcija

$$f_n(t) = \frac{v_0 \cos\theta}{\frac{1}{n}}(1 - e^{-\frac{t}{n}})$$

Vidimo da f_n konvergira uniformno prema $f(t) = v_0 cos\theta \cdot t$

$$\lim_{\alpha \to 0} v_0 cos\theta \cdot e^{-\alpha t} = v_0 cos\theta$$

Promatramo rjesenje po y komponenti te primjenjujemo $\lim_{x \to 0} \frac{e^x - 1}{x} = 1$

$$\lim_{\alpha \to 0}(\frac{v_0 sin\theta}{\alpha} + \frac{g}{\alpha^2})(1 - e^{-\alpha t}) - \frac{g}{\alpha}t = tv_0 \cdot sin\theta$$

$$\lim_{x \to 0}(v_0 sin\theta + \frac{g}{\alpha})e^{-\alpha t} - \frac{g}{\alpha} = v_0 sin\theta - gt$$

kut θ uz koji je x-koordinata točke A maksimalna

b) Pretpostavimo da je $\alpha = \frac{g}{v_0}$. Korištenjem a) djela imamo

$$y(t) = (\frac{v_0^2 sin\theta}{g} + \frac{v_0^2}{g})(1 - e^{-\frac{g}{v_0}t}) - v_0 t$$

U najvišoj točci putanje vrijedi:

$$\dot{y}(t) = 0$$

$$v_0 e^{\frac{-g}{v_0}t}(sin\theta + 1) - v_0 = 0$$

Iz gornje jednadžbe računamo vrijeme t_A, tj. vrijeme potrebno da lopta postigne maksimalnu visinu.

$$t_A = \frac{v_0}{g} ln(sin\theta + 1)$$

uvrštavanjem u $x(t)$ dobivamo visinu najviše točke

$$x(t_A) = \frac{v_0 cos\theta}{\alpha}(1 - e^{-\alpha \frac{v_0 ln(sin\theta + 1)}{g}})$$

Uvrštavanjem $\alpha = \frac{g}{v_0}$ i sređivanjem izraza imamo

$$x(t_A) = \frac{v_0^2 cos\theta}{g}(1 - (sin\theta + 1)^{-1})$$

$$x(t_A) = \frac{v_0^2 cos\theta sin\theta}{g(sin(\theta) + 1)}$$

Problem svodimo na maksimizaciju

$$x(\theta) = \frac{v_0^2 cos\theta sin\theta}{g(sin(\theta) + 1)}$$

Odnosno tražimo rješenje jednadžbe

$$\dot{x}(\theta) = 0$$

$$\dot{x}(\theta) = \frac{v_0^2}{g}\frac{(\cos^2\theta - \sin^2\theta)(\sin\theta + 1) - \cos\theta\sin\theta\cos\theta}{(\sin\theta+1)^2}$$

$$\dot{x}(\theta) = \frac{\left(\cos^2\theta - \sin^2\theta - \sin^2\theta\sin\theta\right)}{(\sin\theta+1)^2} = 0$$

$$(\sin\theta+1)^2 \neq 0 => sin\theta \neq -1$$

$$cos^2\theta - sin^2\theta - sin^3\theta = 0$$

$$1 - 2sin^2\theta - sin^3\theta = 0 => t = sin\theta$$

$$-t^3 - 2t^2 + 1 = 0 => t = \frac{-1 \pm \sqrt{5}}{2}$$

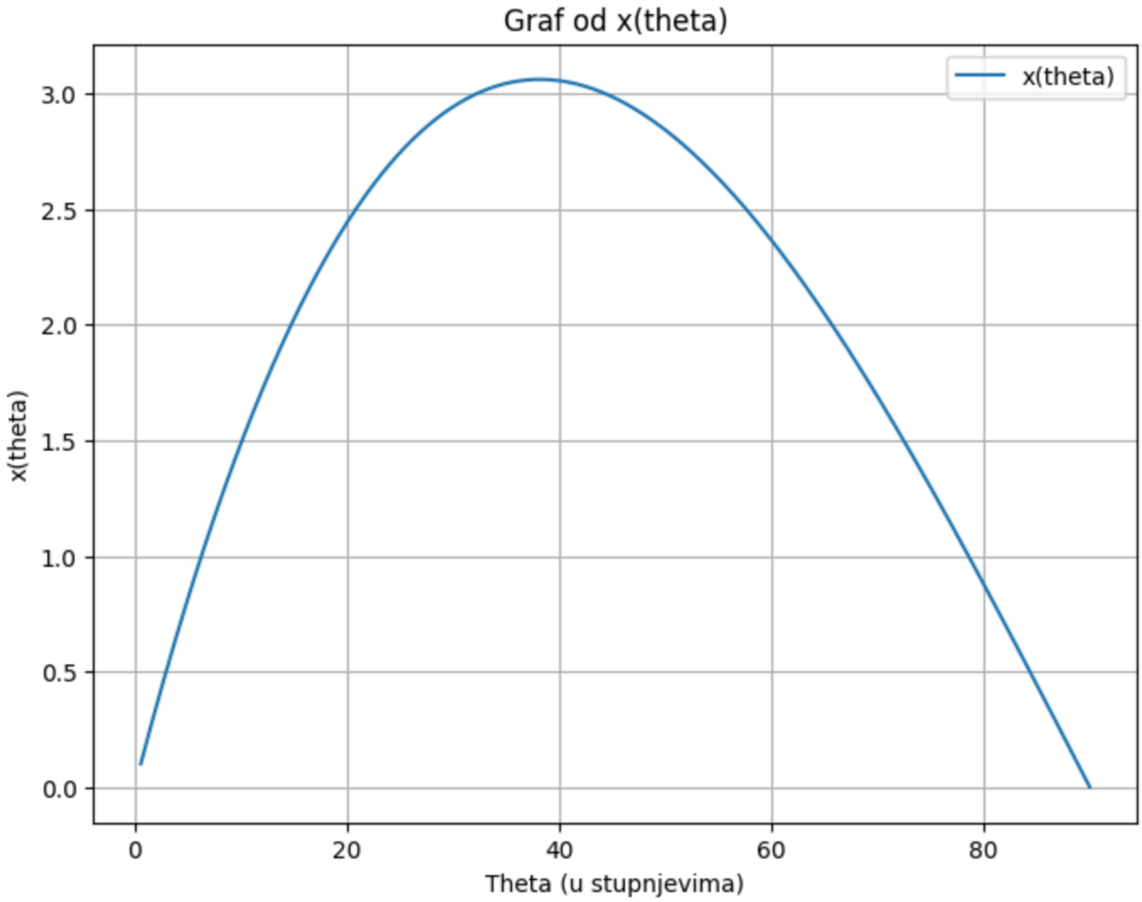
Kako je $\frac{-1-\sqrt{5}}{2} < -1$ slijedi da je jedino rjesenje

$$sin\theta = \frac{-1+\sqrt{5}}{2} => \theta = arcsin(\frac{-1+\sqrt{5}}{2}) \approx 0.6623$$

$\theta \approx 0.6623$ rad, tj. $\theta \approx 38.21$ stupnjeva

Grafički prikaz rješenja

```
Maksimalni theta u radijanima: 0.67
Maksimalni theta u stupnjevima: 38.21
```



Numeričko rješenje zadatka

c)

$$g = 9.81$$

$$v_0 = 20$$

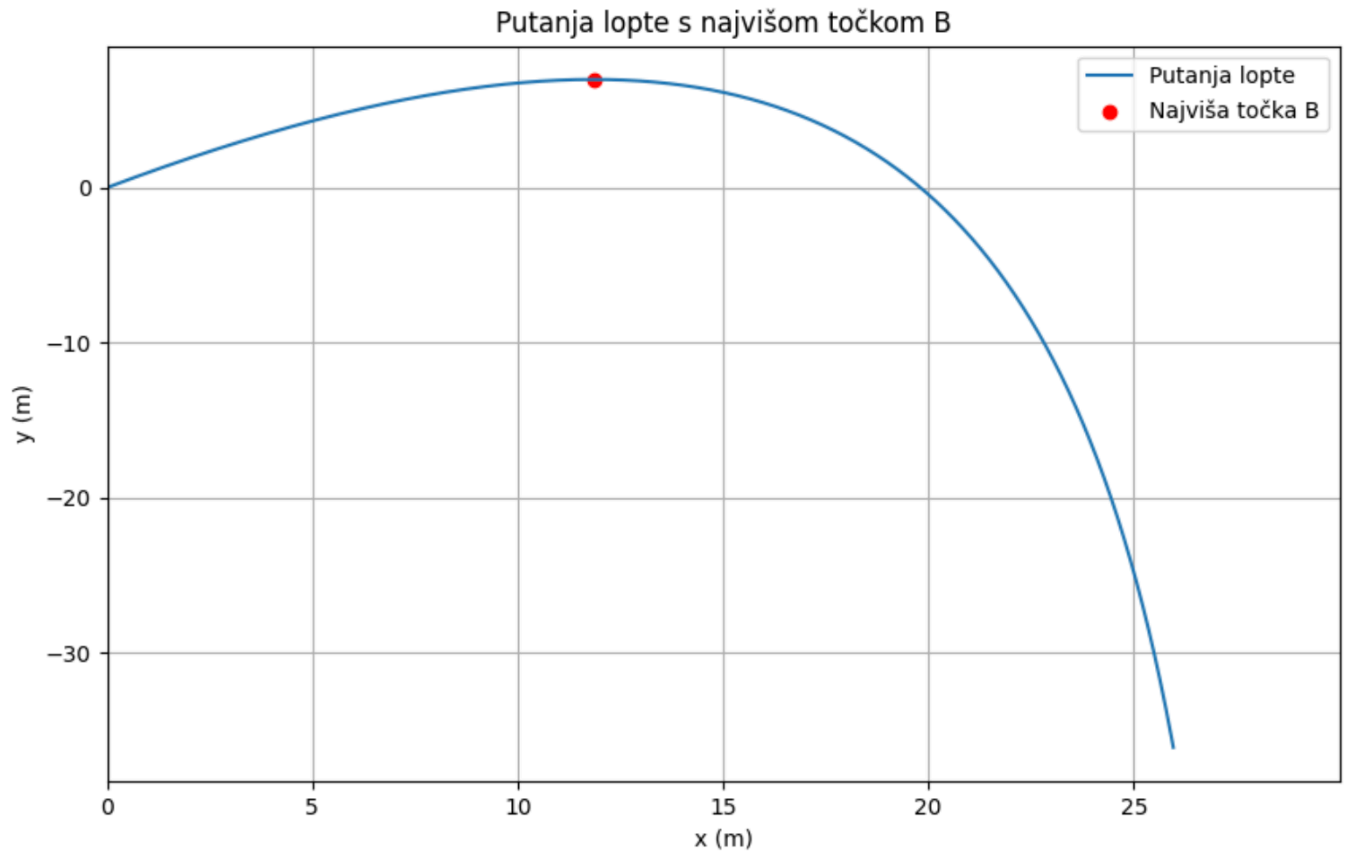
$$\theta = \frac{\pi}{4}$$

$$\alpha = 0.5$$

Objašnjenje koda: Numerički rješavamo zadani ODJ iz a) dijela te dobivamo, kao povratnu vrijednost funkcije odeint, matricu svih vrijednosti $[x, x', y, y']$ u određenom trenutku t. Zatim iz te matice promatramo sve vrijednosti od y i vracamo indeks najveće (to je y kordinata najvišu točku putanje lopte). Uz pomoć indeksa računamo y i x koordinatu te točke B. Rjesenje u kojem sam koristila Events je u file-u najvisa_tocka_putanje.m

[2]:
```
import numpy as np
from scipy.integrate import odeint
import matplotlib.pyplot as plt

g = 9.81
v0 = 20
theta = np.pi / 4
alpha = 0.5

def rhs(p, t, alpha, g):
    x, dxdt, y, dydt = p
    dot_p = [dxdt,
             -alpha * dxdt,
             dydt,
             -alpha * dydt - g]
    return dot_p

p0 = [0, v0 * np.cos(theta), 0, v0 * np.sin(theta)]

n=10000
a_0=0
a_1=5
t = np.linspace(a_0, a_1,n)

sol = odeint(rhs, p0, t, args=(alpha,g)) #odeint je ekvivalent funckije lsode iz Octave


y_values = sol[:, 2]
t_max_index = np.argmax(y_values)
x_max = sol[t_max_index,0]
y_max = y_values[t_max_index]
h=a_1/n
t=a_0+t_max_index*h

print(f"Najviša točka B: ({x_max} m, {y_max} m)")
print(f"Index: {t_max_index:.2f}")
print(f"Vrijeme:{t}")
```

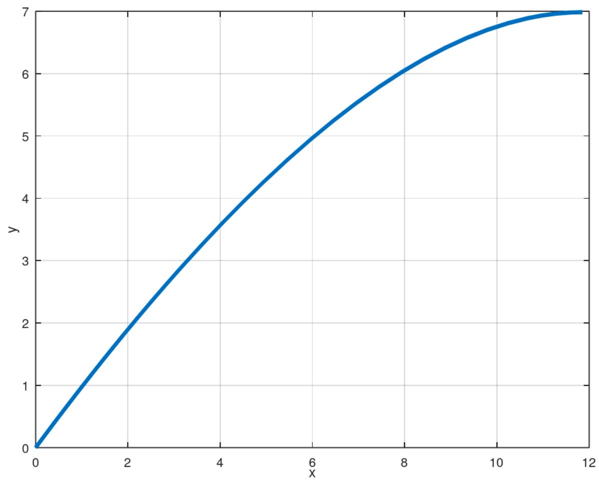
```
Najviša točka B: (11.847818725167652 m, 6.985173114253523 m)
Index: 2171.00
Vrijeme:1.0855000000000001
```



Rješenje u Octave programskom jeziku korištenjem Events

```
% Parametri
g = 9.81;            % ubrzanje sile teže
v0 = 20;             % početna brzina lopte
theta = pi/4;        % kut izbačaja
alpha = 0.5;           % otpor zraka
T = 5
interval=[0,T]       % maksimalna duljina vremenskog intervala


% Početni uvjeti
x1=0;   % Vrijednost prve komponente pocetnog polozaja
x2=0;   % Vrijednost druge komponente pocetnog polozaja
v1=v0 * cos(theta);   % Vrijednost prve komponente pocetne brzine
v2=v0 * sin(theta);   % Vrijednost druge komponente pocetne brzine
uvjeti=[x1 x2 v1 v2]


% Desne strane sustava
F = @(t, y) [y(3); y(4);-alpha*y(3) ; -alpha*y(4) - g];

function [value, isterminal, direction] = najvisa_tocka(t, y)
    value = y(4);               % Pratimo vertikalnu poziciju y(2)
    isterminal = 1;             % Zaustavljamo integraciju kad se događaj ostvari
    direction = -1;             % Promatramo slucajeve samo kad se vrijednost
                                %y(1) smanjuje (1 znaci da
                                %bismo trazili da se vrijednost y(1) povecava, a 0 znaci da nema restrikcija)
end

opts = odeset('RelTol', 1e-11, 'Events', @najvisa_tocka);
[t,y,tevent, yevent, ie]=ode45(F,interval,uvjeti,opts);
% Ispis točke gdje se događaj dogodio
fprintf('Najviša točka putanje postiže se u (x, y) = (%.10f, %.10f)\n', yevent(1, 1), yevent(1, 2));

% Plotanje rezultata
plot(y(:,1), y(:,2), 'LineWidth', 2);
xlabel('x');
ylabel('y');
grid on;
```

```
>> najvisa_tocka_putanje

T = 5
interval =

   0   5

uvjeti =

        0         0   14.1421   14.1421

Najviša točka putanje postiže se u (x, y) = (11.8475917655, 6.9851731350)
```



Određivanje kuta θ

Objasnjenje koda: Definiram funkcije gibanja x(t,theta), y(t,theta) i funkciju koja mjeri udaljenost između lopte i točke B. Problem se svodi na minimizaciju te funkcije korištenjem minimize_scalar, tj. funkcije koja radi isto što i fminbnd

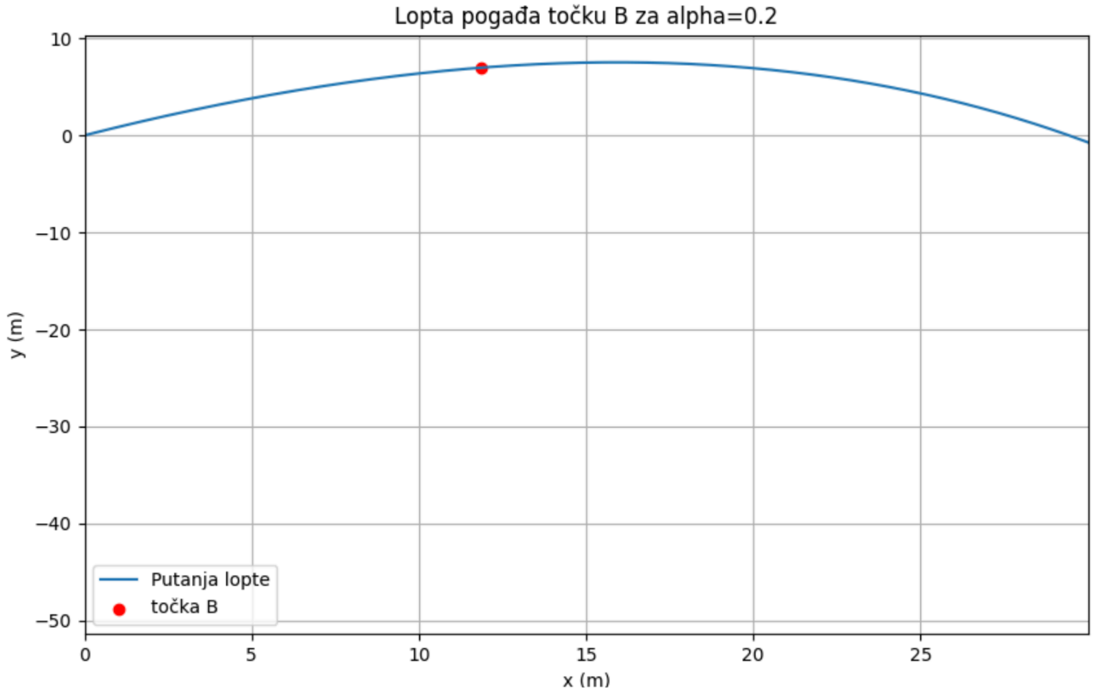
[9]:

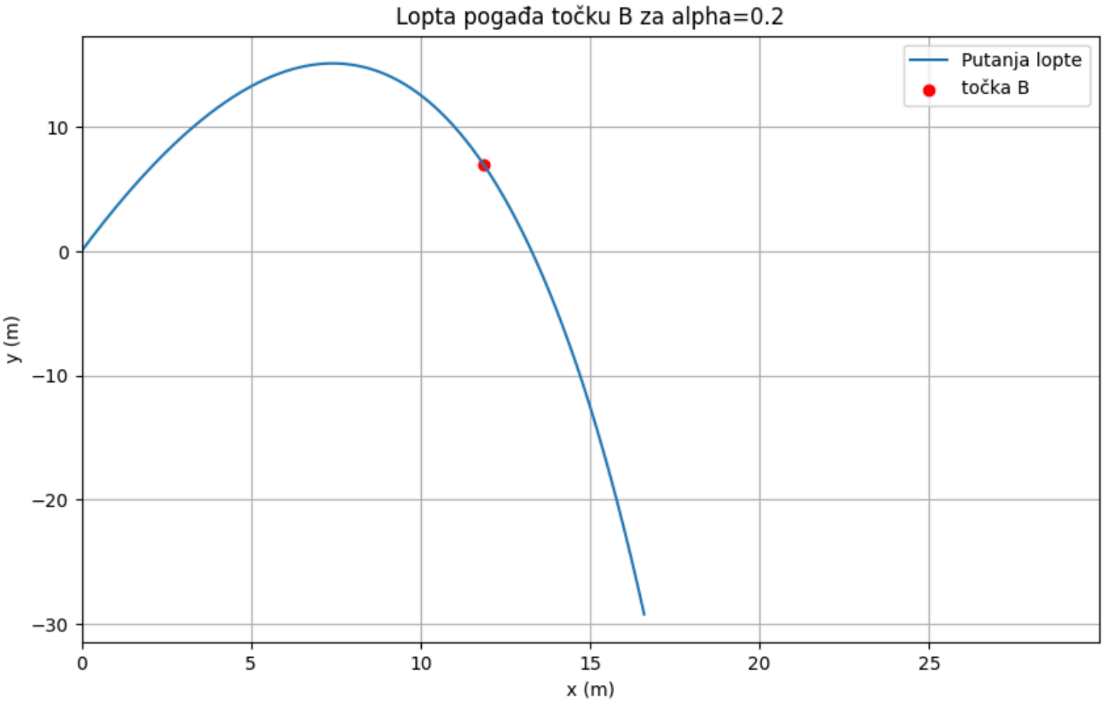
```python
import numpy as np
from scipy.optimize import minimize_scalar

g = 9.81
v0 = 20
alpha = 0.2
x_B = 11.85
y_B = 6.99

# Funkcije gibanja iz a) dijela zadatka u ovisnosti o t i theta
def x_t(t, theta):
    return (v0 * np.cos(theta) / alpha) * (1 - np.exp(-alpha * t))

def y_t(t, theta):
    return ((v0 * np.sin(theta) / alpha) + (g / alpha**2)) * (1 - np.exp(-alpha * t)) - (g / alpha) * t

def distance_to_target(theta):
    a_1 = 5
    a_0=0
    n=1000 #možemo smanjiti grešku povečavanjem broja intervala
    t = np.linspace(a_0, a_1, 1000)

    x = x_t(t, theta)
    y = y_t(t, theta)

    distances = np.sqrt((x - x_B)**2 + (y - y_B)**2)
    min_distance = np.min(distances)

    return min_distance

# minimize_scalar je ekvivalentno funkciji fminbnd iz Octave
result = minimize_scalar(distance_to_target, bounds=(0, np.pi/2), method='bounded')

optimal_theta = result.x
min_distance = result.fun

print(f'Optimalni kut θ: {optimal_theta} radijana ({np.degrees(optimal_theta)} stupnjeva)')
print(f'Greška: {min_distance} metara')

```

```
Optimalni kut θ: 0.7196699452240292 radijana (41.23405050374801 stupnjeva)
Greška: 0.02522551795571367 metara
```

Za određivanje drugog rješenja ograničila sa, funkciju minimize_scakar u gornjem kodu od p/4 do pi/2

```
Optimalni kut θ: 1.3051421703193404 radijana (74.77913802384266 stupnjeva)
Greška: 0.01592695817545975 metara
```





Dodatak: Rješavanje b) dijela u programskom jeziku Octave korištenjem funkcije fminbnd

```
g = 8.91;
v0 = 10;

%trazenje maksimuma pocetne funkcije svodim na trazenje minimuma
%noj negativne funkcije
F = @(theta) -(v0^2 * cos(theta) * (1 - 1/(sin(theta) + 1))) / g;

theta_min = 0;
theta_max = pi/2;
[theta_max, F_max]= fminbnd(F, theta_min, theta_max);
F_max = -F_max;

disp('Theta koje maksimizira funkciju: ');
disp(theta_max)
```

```
>> odredivanje_thete_fminbnd

Theta koje maksimizira funkciju:
0.6662
```