Міністерство освіти і науки України
Дніпровський національний університет імені Олеся Гончара
Факультет прикладної математики
Кафедра комп'ютерних технологій

ЛАБОРАТОРНА РОБОТА №4

Виконавець: студент групи ПК-21м-1
Панасенко Єгор
Сергійович

Постановка задачі

Тема роботи: Побудова синергетичного регулятору для задач оптимального керування (розробка та створення програми).

Мета роботи: Використовуя Maple(Матлаб) отримати чисельний розв'язок регулятора наступної системи диференціальних рівнянь методом АКАР.

Постанова завдання:

Використовуя наступну системи рівнянь Ньютона

$$\begin{aligned} \dot{x}_1(t) &= x_2, \\ \dot{x}_2(t) &= u, \end{aligned} \tag{5.1}$$

побудувати такі нелінійні обернені зв'язки $u(x_1, x_2)$, що дозволяють сформувати у структурі системи (5.1) бажані біфуркації:

1. Біфуркацію типу «сідло-вузол»

$$\dot{x}_{1\psi}(t) = \mu - x_{1\psi}^2(t); \tag{5.2}$$

2. Транскритичну біфуркацію

$$\dot{x}_{1\psi}(t) = \mu x_{1\psi}(t) - x_{1\psi}^2(t); \tag{5.3}$$

3. Біфуркацію типу «вілки»

$$\dot{x}_{1\psi}(t) = \mu x_{1\psi}(t) - x_{1\psi}^3(t). \tag{5.4}$$

В співвідношеннях (5.2)-(5.4) наведено наступне: $x_{1\psi}(t)$ - параметр порядку, μ – відстань до точки біфуркації. Початкові значення до системи (5.1) вибирати самостійно.

Теоретичне введення.

Згідно з методом АКАР, використовуя початкові допустимі значення, система повинна потрапити до околі інваріантної множини $\psi(x_1, x_2) = 0$. Потрібно її визначити таким чином, щоб подальший шлях повз цієї множини мав вигляд типу (5.2), тобто відбулася біфуркація типу «сідло-вузол».

З різних міркувань, потрібно надати макрозміній наступний вигляд

$$\psi(x_1, x_2) = x_2 - \mu + x_1^2. \tag{5.5}$$

Підставляючи тепер (5.5) до функціонально рівняння

$$T\dot{\psi}(t) + \psi = 0 \tag{5.6}$$

з урахуванням системи (5.1) отримуємо, що регулятор має вигляд

$$u(x_1, x_2) = -2x_1x_2 - \frac{\psi}{T}. \tag{5.7}$$

З умови $\psi(x_1, x_2) = 0$ маємо наступне $x_2 = \mu - x_1^2$. Таким чином, після сжаття простору маємо наступне рівняння розвитку

$$\dot{x}_{1\psi}(t) = \mu - x_{1\psi}^2(t);$$

Завдання до виконання:

1. Показати що через час $t = (3{-}4)T$ система (5.1), яка замкнута регулятором вигляду (5.7) вийде на біфуркацію (5.2), яка буде в подальшому визначати її шлях.

2. Виконати побудову регулятору для біфуркацій (5.3) та (5.4).

3. Для усіх випадків представити візуалізацію результатів роботи регуляторів.

Хід роботи

Розглянемо систему диференціальних рівнянь:

$$\begin{bmatrix} \frac{d}{dt}\mathrm{x}_1(t) = \mathrm{x}_2(t) \\ \frac{d}{dt}\mathrm{x}_2(t) = u(\mathrm{x}_1(t), \mathrm{x}_2(t)) \end{bmatrix}$$

Родина стійких екстремалей визначається рівнянням:

$$T u(\mathrm{x}_1(t), \mathrm{x}_2(t), a, T)\frac{d}{d\mathrm{x}_2(t)}\psi(\mathrm{x}_1(t), \mathrm{x}_2(t)) + T\,\mathrm{x}_2(t)\frac{d}{d\mathrm{x}_1(t)}\psi(\mathrm{x}_1(t), \mathrm{x}_2(t)) + \psi(\mathrm{x}_1(t), \mathrm{x}_2(t)) = 0$$

Візмемо:

$$\psi(x_1(t), x_2(t)) = -\mu + {x_1}^2(t) + x_2(t)$$

Підставимо у рівняння:

$$Tu(x_1(t), x_2(t), a, T)\frac{\partial}{\partial x_2(t)}\left(-\mu + {x_1}^2(t) + x_2(t)\right) + T x_2(t)\frac{\partial}{\partial x_1(t)}\left(-\mu + {x_1}^2(t) + x_2(t)\right) - \mu + {x_1}^2(t) + x_2(t) = 0$$

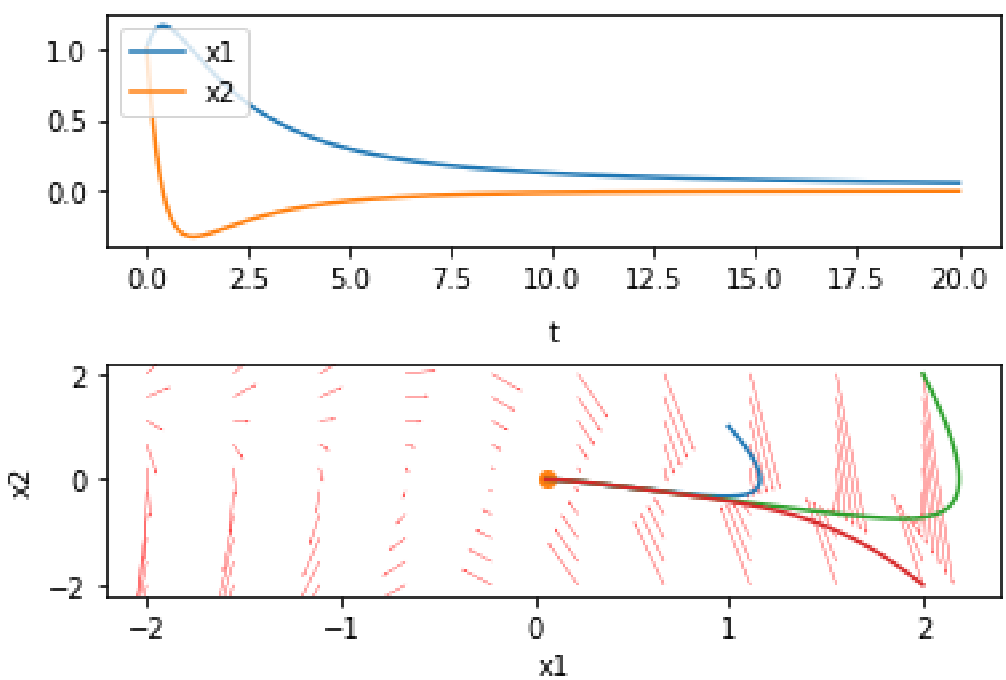
$$Tu(x_1(t), x_2(t), a, T) + 2T x_1(t) x_2(t) - \mu + {x_1}^2(t) + x_2(t) = 0$$

Знайдемо управління:

$$u((x_1(t),\ x_2(t))) = \frac{-2T x_1(t) x_2(t) + \mu - {x_1}^2(t) - x_2(t)}{T}$$

Підставимо управління у систему:

$$\left\{\frac{d}{dt} x_1(t) = x_2(t), \frac{d}{dt} x_2(t) = \frac{-2T x_1(t) x_2(t) + \mu - {x_1}^2(t) - x_2(t)}{T}\right\}$$



Розглянемо систему диференціальних рівнянь:

$$\begin{bmatrix} \frac{d}{dt} x_1(t) = x_2(t) \\ \frac{d}{dt} x_2(t) = u(x_1(t), x_2(t)) \end{bmatrix}$$

Родина стійких екстремалей визначається рівнянням:

$$T u(x_1(t), x_2(t), a, T) \frac{d}{d x_2(t)} \psi(x_1(t), x_2(t)) + T x_2(t) \frac{d}{d x_1(t)} \psi(x_1(t), x_2(t)) + \psi(x_1(t), x_2(t)) = 0$$

Візмемо:

$$\psi(x_1(t), x_2(t)) = -\mu x_1(t) + x_1^2(t) + x_2(t)$$

Підставимо у рівняння:

$$T u(x_1(t), x_2(t), a, T) \frac{\partial}{\partial x_2(t)} \left(-\mu x_1(t) + x_1^2(t) + x_2(t)\right) + T x_2(t) \frac{\partial}{\partial x_1(t)} \left(-\mu x_1(t) + x_1^2(t) + x_2(t)\right) - \mu x_1(t) + x_1^2(t) + x_2(t) = 0$$

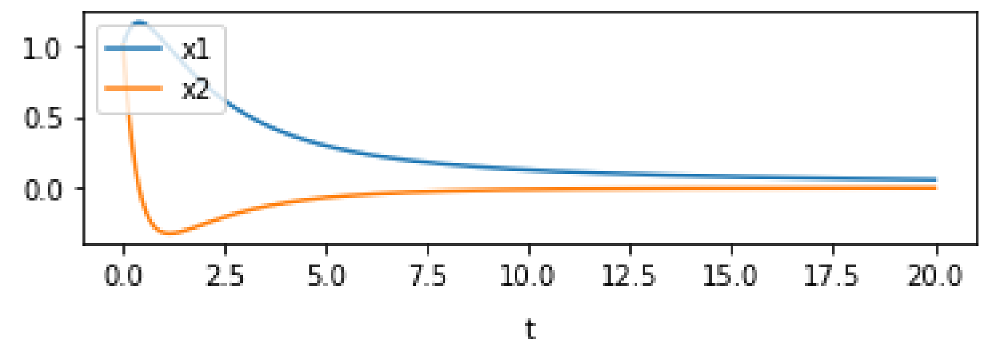
$$T(-\mu + 2 x_1(t)) x_2(t) + T u(x_1(t), x_2(t), a, T) - \mu x_1(t) + x_1^2(t) + x_2(t) = 0$$

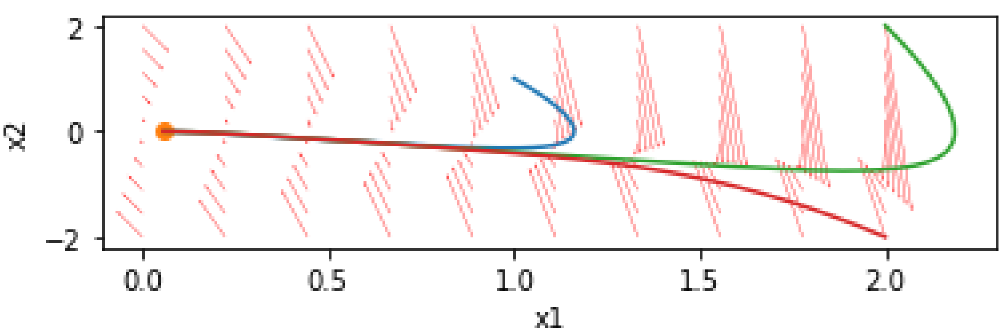
Знайдемо управління:

$$u((x_1(t),\ x_2(t))) = \frac{T(\mu - 2 x_1(t)) x_2(t) + \mu x_1(t) - x_1^2(t) - x_2(t)}{T}$$

Підставимо управління у систему:

$$\left\{\frac{d}{dt} x_1(t) = x_2(t), \frac{d}{dt} x_2(t) = \frac{T(\mu - 2 x_1(t)) x_2(t) + \mu x_1(t) - x_1^2(t) - x_2(t)}{T}\right\}$$





Розглянемо систему диференціальних рівнянь:

$$\begin{bmatrix} \frac{d}{dt} x_1(t) = x_2(t) \\ \frac{d}{dt} x_2(t) = u(x_1(t), x_2(t)) \end{bmatrix}$$

Родина стійких екстремалей визначається рівнянням:

$$Tu(\mathrm{x}_1(t), \mathrm{x}_2(t), a, T)\frac{d}{d\,\mathrm{x}_2(t)}\psi(\mathrm{x}_1(t), \mathrm{x}_2(t)) + T\,\mathrm{x}_2(t)\frac{d}{d\,\mathrm{x}_1(t)}\psi(\mathrm{x}_1(t), \mathrm{x}_2(t)) + \psi(\mathrm{x}_1(t), \mathrm{x}_2(t)) = 0$$

Візмемо:

$$\psi(\mathrm{x}_1(t), \mathrm{x}_2(t)) = -\mu\,\mathrm{x}_1(t) + \mathrm{x}_1{}^3(t) + \mathrm{x}_2(t)$$

Підставимо у рівняння:

$$Tu(\mathrm{x}_1(t), \mathrm{x}_2(t), a, T)\frac{\partial}{\partial\,\mathrm{x}_2(t)}\left(-\mu\,\mathrm{x}_1(t) + \mathrm{x}_1{}^3(t) + \mathrm{x}_2(t)\right) + T\,\mathrm{x}_2(t)\frac{\partial}{\partial\,\mathrm{x}_1(t)}\left(-\mu\,\mathrm{x}_1(t) + \mathrm{x}_1{}^3(t) + \mathrm{x}_2(t)\right) - \mu\,\mathrm{x}_1(t) + \mathrm{x}_1{}^3(t) + \mathrm{x}_2(t) = 0$$

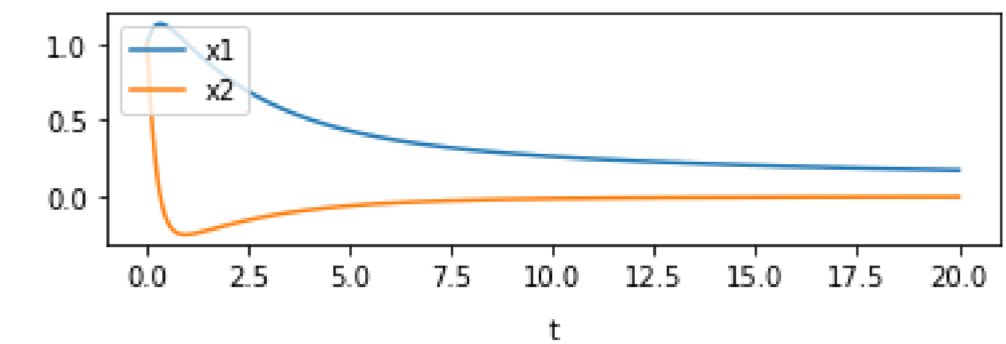
$$T\left(-\mu + 3\,\mathrm{x}_1{}^2(t)\right)\mathrm{x}_2(t) + Tu(\mathrm{x}_1(t), \mathrm{x}_2(t), a, T) - \mu\,\mathrm{x}_1(t) + \mathrm{x}_1{}^3(t) + \mathrm{x}_2(t) = 0$$

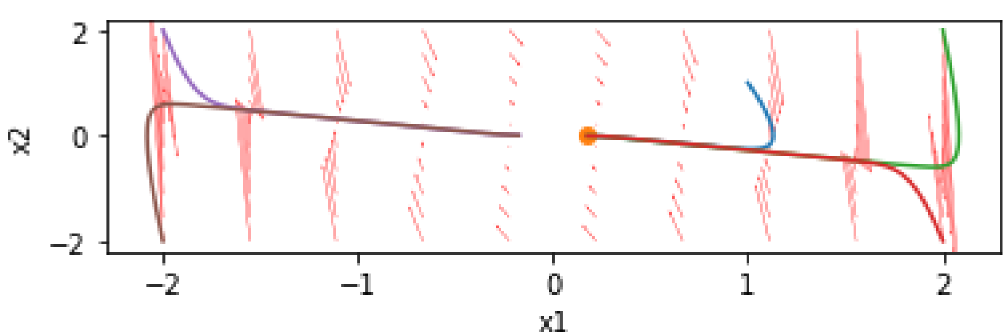
Знайдемо управління:

$$u((\mathrm{x}_1(t),\ \mathrm{x}_2(t))) = \frac{T\left(\mu - 3\,\mathrm{x}_1{}^2(t)\right)\mathrm{x}_2(t) + \mu\,\mathrm{x}_1(t) - \mathrm{x}_1{}^3(t) - \mathrm{x}_2(t)}{T}$$

Підставимо управління у систему:

$$\left\{\frac{d}{dt}\mathrm{x}_1(t) = \mathrm{x}_2(t), \frac{d}{dt}\mathrm{x}_2(t) = \frac{T\left(\mu - 3\,\mathrm{x}_1{}^2(t)\right)\mathrm{x}_2(t) + \mu\,\mathrm{x}_1(t) - \mathrm{x}_1{}^3(t) - \mathrm{x}_2(t)}{T}\right\}$$





Код програми

```
#!/usr/bin/env python3

from sympy import *
from sympy.solvers.ode.ode import odesimp
import matplotlib.pyplot as plt
import numpy as np
from scipy.integrate import odeint
from IPython.display import display, Latex
init_printing(use_latex='mathjax')
pprint = display

t, a, T, mu = symbols("t, a, T, mu")
x1, x2, u, psi = symbols("x1, x2, u, psi", cls=Function)
x1, x2 = x1(t), x2(t)
psi = psi(x1, x2)
x = (x1, x2)

def sys(x1, x2, t, u, a, T):
    return [x2, u(x1, x2, a, T)]

ss = sys(x1, x2, t, u, a, T)

def find_x_sol(sys, psi_fun):
    print("Розглянемо систему диференціальних рівнянь:")
    pprint(Matrix([Eq(diff(x1,t), x2), Eq(diff(x2,t), u(x1, x2))]))
    print("Родина стійких екстремалей визначається рівнянням:")
    eq = Eq(T*diff(psi, x1)*ss[0]+T*diff(psi, x2)*ss[1] + psi, 0)
    pprint(eq)
    print("Візмемо:")
    pprint(Eq(psi, psi_fun))
    eq = eq.subs(psi, psi_fun)
    print("Підставимо у рівняння:")
    pprint(eq)
    eq = simplify(eq)
    pprint(eq)
    sol = solve(eq, u(x1, x2, a, T))[0]
    print("Знайдемо управління:")
    pprint(Eq(u(x), sol))
    u_sol = lambdify([x1, x2, a, T], sol, modules='sympy')
    sol = sys(x1, x2, 0, u_sol, a, T)
    print("Підставимо управління у систему:")
    pprint(set([Eq(diff(x[i],t), sol[i]) for i in range(2)]))
    x_sol = lambdify(((x1, x2), t, a, T, mu), sol, modules='sympy')
    return x_sol


def pp(f, r1=(0.0, 2.0), r2=(-2.0, 2.0)):
    y1 = np.linspace(*r1, 10)
    y2 = np.linspace(*r2, 10)
    Y1, Y2 = np.meshgrid(y1, y2)
    u, v = np.zeros(Y1.shape), np.zeros(Y2.shape)
    NI, NJ = Y1.shape
    for i in range(NI):
        for j in range(NJ):
            yprime = f([Y1[i, j], Y2[i, j]], 0, 1, 1, 0)
            u[i,j] = yprime[0]
            v[i,j] = yprime[1]
    return Y1, Y2, u, v

# ~ u_sol = find_u_sys(sys, x)

tspan = np.linspace(0, 20, 1000)
```

```
x_sol = find_x_sol(sys, x2 - mu + x1**2)
res = odeint(x_sol, [1, 1], tspan, args=(1, 1, 0))

plt.figure(1)
fig, ax = plt.subplots(2, 1)
fig.subplots_adjust(hspace=0.5, wspace=0.5)
ax[0].plot(tspan, res, label=["x1", "x2"])
ax[0].set_xlabel("t", labelpad=10)
ax[0].legend(loc='upper left')
ax[1].plot(res[:,0], res[:,1])
ax[1].plot(res[-1,0], res[-1,1], "o")
res = odeint(x_sol, [2, 2], tspan, args=(1, 1, 0))
ax[1].plot(res[:,0], res[:,1])
res = odeint(x_sol, [2, -2], tspan, args=(1, 1, 0))
ax[1].plot(res[:,0], res[:,1])
ax[1].quiver(*pp(x_sol, (-2, 2)), color='r', width=0.001)
ax[1].set_xlabel("x1")
ax[1].set_ylabel("x2")
plt.show(block=False)


x_sol = find_x_sol(sys, x2 - mu*x1 + x1**2)
res = odeint(x_sol, [1, 1], tspan, args=(1, 1, 0))

plt.figure(2)
fig, ax = plt.subplots(2, 1)
fig.subplots_adjust(hspace=0.5, wspace=0.5)
ax[0].plot(tspan, res, label=["x1", "x2"])
ax[0].set_xlabel("t", labelpad=10)
ax[0].legend(loc='upper left')
ax[1].plot(res[:,0], res[:,1])
ax[1].plot(res[-1,0], res[-1,1], "o")
res = odeint(x_sol, [2, 2], tspan, args=(1, 1, 0))
ax[1].plot(res[:,0], res[:,1])
res = odeint(x_sol, [2, -2], tspan, args=(1, 1, 0))
ax[1].plot(res[:,0], res[:,1])
ax[1].quiver(*pp(x_sol), color='r', width=0.001)
ax[1].set_xlabel("x1")
ax[1].set_ylabel("x2")
plt.show(block=False)


x_sol = find_x_sol(sys, x2 - mu*x1 + x1**3)
res = odeint(x_sol, [1, 1], tspan, args=(1, 1, 0))

plt.figure(3)
fig, ax = plt.subplots(2, 1)
fig.subplots_adjust(hspace=0.5, wspace=0.5)
ax[0].plot(tspan, res, label=["x1", "x2"])
ax[0].set_xlabel("t", labelpad=10)
ax[0].legend(loc='upper left')
ax[1].plot(res[:,0], res[:,1])
ax[1].plot(res[-1,0], res[-1,1], "o")
res = odeint(x_sol, [2, 2], tspan, args=(1, 1, 0))
ax[1].plot(res[:,0], res[:,1])
res = odeint(x_sol, [2, -2], tspan, args=(1, 1, 0))
ax[1].plot(res[:,0], res[:,1])
res = odeint(x_sol, [-2, 2], tspan, args=(1, 1, 0))
ax[1].plot(res[:,0], res[:,1])
res = odeint(x_sol, [-2, -2], tspan, args=(1, 1, 0))
ax[1].plot(res[:,0], res[:,1])
ax[1].quiver(*pp(x_sol, (-2, 2)), color='r', width=0.001)
```

```
ax[1].set_xlabel("x1")
ax[1].set_ylabel("x2")
plt.show()
```