

ЕВОЛЮЦІЙНА МОДЕЛЬ ОПТИМІЗАЦІЇ ДИНАМІЧНИХ ХАРАКТЕРИСТИК АВТОМАТИЧНОГО ПОТЕНЦІОМЕТРА

© Павельчак А. Г., Самотий В. В., Баклан Ю. В., 2015

Розглянуто параметричну оптимізацію для потенціометричної слідкувальної системи з використанням генетичного алгоритму. Отримано якісні характеристики перехідного процесу системи керування.

Ключові слова: оптимізація, генетичний алгоритм, система керування.

Parametric optimization of potentiometric tracking system was considered. Optimization was conducted using Genetic Algorithm. Qualitative characteristics of the transition process of control system were obtained.

Key words: optimization, genetic algorithm, control system.

Вступ

Проблема оптимізації динамічних характеристик технічних систем посідає чільне місце в наукових дослідженнях, інженерії, економіці, управлінні, промисловості та інших областях. Вона орієнтована на вибір кращих з багатьох можливих рішень в умовах реальних режимів роботи, а також побудову обчислювальних методів для пошуку оптимальних рішень, дослідження аналітичних залежностей та вивчення продуктивності числових алгоритмів.

Завдяки швидкому розвитку високопродуктивних комп'ютерів та створенню нових ефективних обчислювальних методів ми отримали можливість вивчати та розв'язати все складніші оптимізаційні задачі. В цьому переліку слід зазначити і задачі математичного моделювання оптимальних систем автоматичного керування.

Розглянемо параметричну оптимізацію для слідкувальної системи автоматичного потенціометра з використанням генетичного алгоритму.

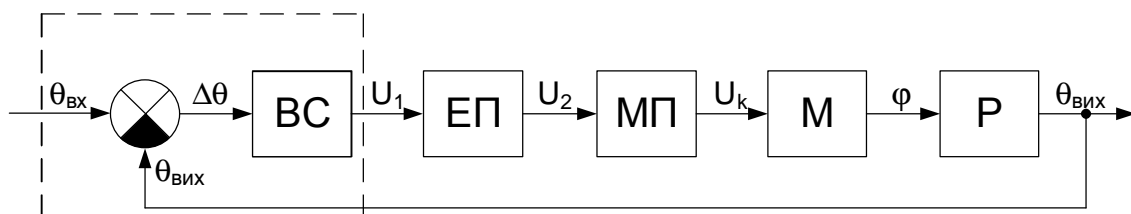


Рис. 1. Структурна схема системи керування автоматичним потенціометром

Автоматичний потенціометр (рис. 1) – це слідкувальна система автоматичного керування (САК), яка дає можливість на значній відстані керувати кутовим положенням об'єкта. До класу слідкувальних систем належать САК, у яких керована величина повинна слідувати за задавальною величиною, яка змінюється за невідомим законом. Для числового тестування задавальну величину

вибирають відомою часовою функцією, як правило, одиничним стрибком. Задавальною величиною для цієї системи є кут θ_{ex} . Вихідною керованою величиною є кут $\theta_{вих}$. Тобто, ми можемо на значній відстані змінювати кутове положення $\theta_{вих}$ зміною задавального кута θ_{ex} . У наведеній системі керування давачами кутових положень θ_{ex} і $\theta_{вих}$ є потенціометри. Вони включені в мостову схему, одна з діагоналей якої живиться постійною напругою U_{on} , а інша діагональ утворена повзунками роторів потенціометрів (рис. 2). За рівності кутових положень роторів потенціометрів напруга U_1 на виході вимірювальної схеми (ВС) дорівнюватиме нулю. При зміні кутового положення ротора задавального потенціометра на вході електронного підсилювача (ЕП) з'явиться постійна напруга U_1 , знак якої визначить напрямок переміщення задаючого кута θ_{ex} , і амплітуда напруги буде пропорційна величині кута θ_{ex} . Підсилений сигнал U_2 подається на вхід двотактного (реверсного) магнітного підсилювача (МП) з виходом на постійному струмі, який виконує роль драйвера мотора постійного струму (М). У цій САК використано мотор постійного струму із незалежним збудженням. Ротор мотора через редуктор (Р) пов'язаний з об'єктом керування (ротором потенціометричного давача). Внаслідок реалізації від'ємного зворотного зв'язку здійснюватиметься слідування вихідного кута $\theta_{вих}$ за задавальним кутом θ_{ex} . У САК реалізовано принцип регулювання за відхиленням. Ця система є системою непрямого регулювання. За рахунок використання мотора постійного струму в системі введено інтегральний закон керування, тому що кут вихідного валу φ є інтегралом від кутової швидкості ω за часом.

Застосування оптимізації до цієї САК дасть можливість підібрати параметри розроблюваної системи, які забезпечать якісні динамічні характеристики, а також мінімізують статичну похибку, яка вноситься особливостями роботи мотора постійного струму.

Аналіз питання

Методи оптимізації можна умовно поділити на комбінаторні та безперервні [1].

Для комбінаторної оптимізації ми використовуємо точні методи для пошуку глобального оптимуму (для цього вони перебирають розв'язки у просторі пошуку). Наприклад, метод гілок і меж (Branch and bound), пошук у просторі станів (State space search), метод Гоморі (Cutting-plane method) та інші [2, 3]. Ці методи дають можливість знайти точний глобальний оптимум для певних типів задач. Однак їх неможливо застосувати для задач, що мають дуже широкий простір пошуку. Наближені методи пошуку вже використовують для складних задач. Для таких задач ми можемо зробити вибір між "спеціалізованими" евристичними методами, що мають відношення лише до цієї конкретної задачі (наприклад, перший кращий алгоритм для задачі упакування контейнера [4]) та метаевристичними методами.

Серед метаевристичних методів розрізняють такі, що працюють в певному околі (покращують одне рішення за один раз), та розосереджені методи, які опрацьовують паралельно цілу популяцію рішень (генетичні алгоритми тощо) [5–9]. Метаевристичні методи побудовані на загальній ідеї (імітація природного процесу, наприклад, відпал металу [5]). Їх можна адаптувати до всіх видів задач із задовільним результатом.

Для безперервної оптимізації можна легко відокремити лінійну задачу (яка, наприклад, може бути оброблена методами лінійного програмування) від нелінійної, де можна виявити складну структуру оптимізації. У цьому випадку прагматичним рішенням може бути використання локальних методів, які або використовують, або не використовують похідні цільової функції. Перші методи використовують похідні цільової функції для пошуку оптимуму. Коли ця техніка застосовується для загальної задачі безперервної оптимізації, то не завжди можна віднайти глобальний оптимум. Тоді знаходять лише локальний оптимум. В окреме сімейство об'єднуються різноманітні методи, що не використовують похідних цільових функцій. Серед них можна зазначити конструктивні та стохастичні методи. Конструктивні методи: будуємо розв'язок однієї змінної за іншою, та при цьому забороняємо модифікацію вже модифікованої змінної. Наприклад,

жадібний метод (Greedy method) належить до сімейства конструктивних методів. Стохастичні методи побудовані на випадковому процесі. Наприклад, метод випадкового блукання (Random walk method), за яким ми випадково вибираємо розв'язок в околі поточного рішення, і цей розв'язок стає поточним незалежно від значення цільової функції.

Якщо є дуже багато локальних мінімумів, необхідно використовувати глобальний метод: тоді використовуємо метаевристику, що є доброю альтернативою класичним глобальним методам оптимізації (які потребують певних математичних властивостей відносно цільової функції). Для глобального пошуку можна використовувати і класичні методи, які вважаються фундаментальними методами в оптимізації. Тут можна виділити метод градієнтного спуску, який для опуклої функції дає можливість знайти глобальний оптимум.

Гібридні методи часто комбінують метаевристичні та локальні методи пошуку. Локальні методи пошуку при цьому дають можливість покращувати поточні рішення.

Постановка задачі

Проектування еволюційної моделі оптимізації динамічних характеристик автоматичного потенціометра передбачає такі етапи:

1. Розроблення математичної моделі потенціометричної слідкуючої системи. Рівняння стану моделі системи керування мають бути записані в нормальній формі Коші.

2. Аналіз та алгоритмічна реалізація генетичного алгоритму з представленням генів у дійсних числах. Алгоритмічна реалізація повинна забезпечувати таку функціональність: довільний вибір основних операторів генетичного алгоритму, вибір діапазонів значень параметрів, кількості хромосом в популяції, ймовірності мутації, кількості поколінь, можливості призупинення та відновлення роботи генетичного алгоритму, продовження пошуку оптимуму при досягненні встановленої кількості поколінь, а також збереження у файл на диску отриманої кінцевої популяції та відновлення її з файла для продовження пошуку оптимуму. Реалізація алгоритму повинна давати можливість призупиняти процес пошуку оптимуму, змінювати значення параметрів генетичного алгоритму та продовжувати далі роботу над пошуком оптимуму.

3. Пошук оптимуму для потенціометричної слідкуючої системи. Якісним критерієм для оптимізації системи виберемо реакцію системи на зміну вхідного сигналу, тобто будемо оптимізувати якість перехідного процесу в системі керування.

Математична модель системи керування

Модель для нашої системи керування автоматичним потенціометром (рис. 2) складається з рівнянь послідовно підключених блочних елементів системи, де вихідні сигнали одних елементів є вхідними сигналами для наступних елементів. При цьому, враховуючи зворотний зв'язок, перший блок (вимірювальна схема) містить як вхідний, так і вихідний сигнали системи керування.

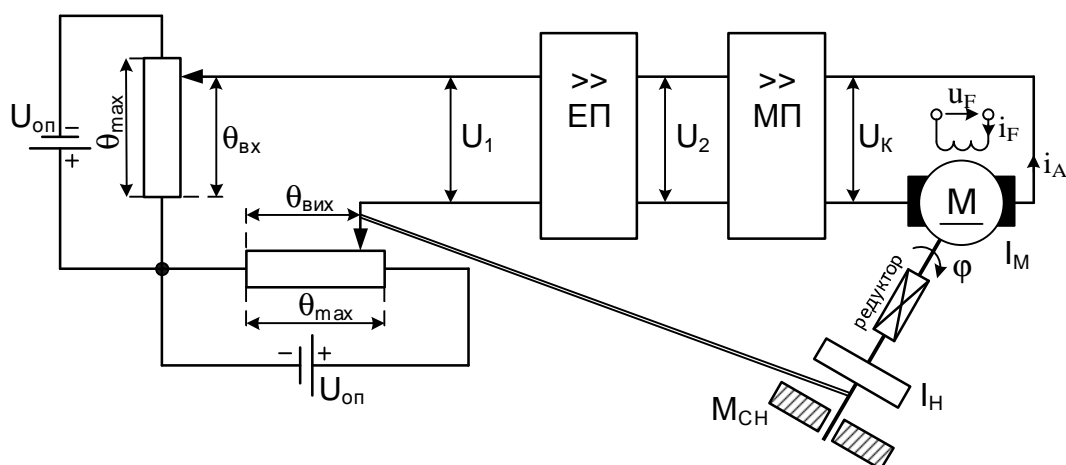


Рис. 2. Система керування автоматичним потенціометром

Елементи системи керування описуються такими рівняннями.

1. Вимірвальна схема:

$$U_1 = \frac{U_{on}}{\theta_{max}} \theta_{ex} - \frac{U_{on}}{\theta_{max}} \theta_{вих}, \quad (1)$$

де U_{on} – опорна напруга, θ_{max} – максимальний кут повороту потенціометричних давачів.

2. Електронний підсилювач:

$$U_2 = f(U_1). \quad (2)$$

Залежність (2) апроксимована таким виразом

$$U_2 = \begin{cases} K_{ЕП} \cdot U_1 & \text{якщо } |U_1| \leq B_x \\ S_3(U_1) & \text{якщо } B_x < |U_1| < C_x, \\ \text{sign}(U_1) \cdot U_{maxEP} & \text{якщо } |U_1| \geq C_x \end{cases} \quad (3)$$

де $K_{ЕП}$ – коефіцієнт підсилення електронного підсилювача; U_{maxEP} – напруга насичення електронного підсилювача; $S_3(U_1)$ – кубічний сплайн, розрахункові точки якого мають такі значення (рис. 3): $B_y = \frac{1}{2} \cdot U_{maxEP}$,

$$B_x = B_y / K_{ЕП}, \quad C_y = U_{maxEP}, \quad C_x = 3 \cdot B_x.$$

3. Магнітний двотактний підсилювач з виходом на постійному струмі:

$$T_M \frac{dU_k}{dt} + U_k = f(U_2), \quad (4)$$

де T_M – стала часу магнітного підсилювача. Залежність $f(U_2)$ апроксимована таким виразом:

$$f(U_2) = \begin{cases} K_{МП} \cdot U_2 & \text{якщо } |U_2| \leq B_x \\ S_3(U_2) & \text{якщо } B_x < |U_2| < C_x, \\ \text{sign}(U_2) \cdot U_{maxMP} & \text{якщо } |U_2| \geq C_x \end{cases}, \quad (5)$$

де $K_{МП}$ – коефіцієнт підсилення магнітного підсилювача; U_{maxMP} – напруга насичення магнітного підсилювача; $S_3(U_2)$ – кубічний сплайн, розрахункові точки якого мають такі значення (рис. 3): $B_y = \frac{1}{2} \cdot U_{maxMP}$, $B_x = B_y / K_{МП}$, $C_y = U_{maxMP}$, $C_x = 3 \cdot B_x$.

4. Мотор постійного струму із незалежним збудженням:

$$\frac{di_A}{dt} = S_A \cdot U_k + T_A \cdot i_F + E_A, \quad \frac{di_F}{dt} = T_F \cdot U_k + S_F \cdot i_F + E_F, \quad (6)$$

де $S_A = 1/(L_A + L_{AF}L_{FA}/L_{FF})$, $T_A = -S_A L_{AF}/L_{FF}$, $T_F = -S_A L_{FA}/L_{FF}$,

$$E_F = -(L_{FA} \cdot E_A + r_F \cdot i_F)/L_{FF}, \quad S_F = (1 - L_{FA} \cdot T_A)/L_{FF},$$

$$E_A = S_A (L_{AF} \cdot r_F \cdot i_F / L_{FF} - c \cdot \omega \cdot \Phi - \Delta u - r_A \cdot i_A).$$

Позначення величин: L_A – сумарна індуктивність послідовного кола ротора; L_{FF} – індуктивність обмотки збудження; L_{AF} , L_{FA} – взаємні індуктивності кола ротора і кола збудження; r_A , r_F – активні опори кола ротора і кола збудження; ω – кутова швидкість обертання ротора МПС; Φ – магнетний потік мотора; c – конструктивна стала ротора МПС; Δu – падіння напруги в щітковому контакті.

У скомпенсованих моторах врахувати насичення магнетопровідника можна наближено за кривою намагнення $\Phi = \Phi(i_F)$, $L_{FF} = L_{FF}(i_F)$. У ненасиченому моторі $\Phi = ki_F$, $L_{FF} = const$.

Рівняння руху записують за рівнянням Даламбера

$$\frac{d\omega}{dt} = (c \cdot \Phi \cdot i_A - \text{sign}(\omega) \cdot M_O) / J, \quad (7)$$

де J – приведений момент інерції; M_O – момент опору,

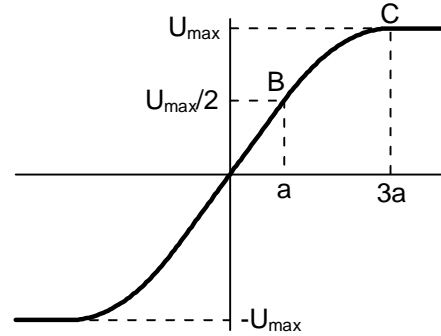


Рис. 3. Графік апроксимованих кривих для залежностей (3) та (5)

$$J = I_M + \frac{I_H}{K_{ред}^2}, \quad M_O = \frac{M_{CH}}{K_{ред}},$$

де I_M – момент інерції ротора мотора; I_H – момент інерції навантаження; M_{CH} – статичний об'єкт навантаження; $K_{ред}$ – коефіцієнт редукції.

Під час моделювання роботи мотора постійного струму важливою є умова старту обертання мотора (8):

$$|c \cdot \Phi \cdot i_A| > M_O. \quad (8)$$

Допоки умову (8) не виконали, встановлюємо значення $\omega = 0$.

Кут повороту знайдемо за рівнянням кутової швидкості

$$\frac{d\varphi}{dt} = \omega. \quad (9)$$

5. Редуктор:

$$\varphi = K_{ред} \cdot \theta_{вих}. \quad (10)$$

Виконавши приведення рівнянь (4, 6, 7, 9) до нормальної форми Коші, можемо проінтегрувати модель системи керування та отримати якісну характеристику її перехідного процесу.

Реалізація генетичного алгоритму

Параметрична оптимізація нашої системи керування здійснюється за допомогою класичного генетичного алгоритму з представленням генів дійсними числами [6, 7]. З метою спрощення структури генетичного алгоритму та його універсальності вибрані параметри об'єкта для оптимізації повинні бути нормалізованими в межах [0, 1]. Кодування параметрів в гени та їхнє декодування здійснюють за формулами (11, 12):

$$gene_i = \frac{param_i - valueMin_i}{valueMax_i - valueMin_i}; \quad (11)$$

$$param_i = valueMin_i + gene_i \cdot (valueMax_i - valueMin_i). \quad (12)$$

У нашій алгоритмічній реалізації хромосома представлена масивом генів з типом даних у вигляді чисел з плаваючою комою подвійної точності.

Розроблена нами реалізація генетичного алгоритму має таку послідовність дій:

1. *Створення початкової популяції з ініціалізацією хромосом та обчисленням фітнес-значень.* Гени початкової популяції генеруються випадковим чином та є нормалізованими в межах [0, 1]. Обчислення фітнес-значень для хромосом розпаралелюється залежно від кількості фізичних (логічних) ядер процесора персонального комп'ютера.

2. *Сортування популяції за фітнес-значенням.* Виконується за кращими фітнес-значеннями (мінімальними чи максимальними). Краще фітнес-значення при цьому знаходиться внизу популяції (останнє значення в масиві хромосом), гірше фітнес-значення – на початку популяції.

3. *Селекція.* Відбираємо хромосоми для популяції за допомогою лінійного ранжування:

$$Fitne\beta(Pos) = 2 - SP + 2 \cdot (SP - 1) \cdot \frac{Pos - 1}{Nind - 1}, \quad (13)$$

де $Nind$ – кількість хромосом (особин) в популяції; Pos – позиція хромосоми в популяції (найменш пристосовані хромосоми мають $Pos=1$, найбільш пристосовані – $Pos=Nind$); SP – коефіцієнт тиску селекції, який може набувати значення в межах [1,0; 2,0]. Згідно з отриманим значенням $Fitne\beta$ визначається кількість входжень хромосоми до популяції. Додаткові копії кращих хромосом вставляємо із заміною гірших на початок популяції. При тестуванні генетичного алгоритму коефіцієнт SP виявився оптимальним, починаючи зі значення 1,6.

4. *Відбір елітних хромосом.* Копіюємо задану кількість кращих хромосом у проміжний буфер.

5. *Спаровування*. Для цієї операції ми створили декілька операторів.

a. *Кращі з кращими*: у відсортованій популяції спаровування відбувається за ранжиrom: спершу попарно кращі, а потім поступово гірші.

b. *Кращі з гіршими*: у відсортованій популяції відбираються хромосоми для спаровування з протилежних кінців списку, і так поступово до середини списку популяції.

c. *Випадковий відбір*: за допомогою генератора випадкових чисел генерується ціле число в межах розмірності кількості хромосом у популяції та за цим порядковим значенням хромосома витягується зі списку популяції, після цього розмірність зменшується на одиницю; далі аналогічно вибирається друга хромосома і т.д. За цією стратегією хромосоми зі списку популяції беруть участь у схрещуванні лише один раз.

d. *Відбір перемішуванням*: у дві групи випадковим чином розкидаються хромосоми з популяції, далі хромосоми в цих групах сортуються за їхніми фітнес-значеннями; хромосоми для утворення пар вибирають за їхнім рангом у підгрупах, тобто кращі з кращими, гірші з гіршими.

e. *Половина на половину*: відсортовану популяцію ділять навпіл, потім пари утворюються за рангом у підгрупах: кращі з однієї підгрупи з кращими з іншої підгрупи, а гірші з гіршими.

Для практичної реалізації оптимальним виявився оператор “кращі з кращими”, а при наближенні до кінця пошуку при виродженні популяції оператор “кращі з гіршими”.

6. *Схрещування*. Використовуємо оператор, що імітує бінарне одноточкове схрещування [7]. Нехай ми маємо пару батьків для схрещування

$$\begin{aligned} parent_1 &= \{p_{m1}, p_{m2}, \dots, p_{mr}, \dots, p_{mNgenes}\} \\ parent_2 &= \{p_{f1}, p_{f2}, \dots, p_{fr}, \dots, p_{fNgenes}\} \end{aligned} \quad (14)$$

де m та f – символи, що розрізняють гени мами та тата. Генеруємо випадкове ціле число r в межах розмірності кількості генів у хромосомі. Так отримаємо номер гена, що буде точкою схрещування. Потім вибрані гени модифікують за такими формулами

$$\begin{aligned} p_{new1} &= p_{mr} - \beta \cdot (p_{mr} - p_{fr}) \\ p_{new2} &= p_{fr} + \beta \cdot (p_{mr} - p_{fr}) \end{aligned} \quad (15)$$

де β – випадкове число в межах $[0, 1]$. Утворені внаслідок схрещування нащадки матимуть вигляд:

$$\begin{aligned} offspring_1 &= \{p_{m1}, p_{m2}, \dots, p_{new1}, \dots, p_{fNgenes}\} \\ offspring_2 &= \{p_{f1}, p_{f2}, \dots, p_{new2}, \dots, p_{mNgenes}\} \end{aligned} \quad (16)$$

7. *Обчислення фітнес-значення для отриманих нащадків після схрещування*. Обчислення фітнес-значення розпаралелюється залежно від кількості фізичних (логічних) ядер процесора персонального комп'ютера.

8. *Сортування популяції за фітнес-значенням*. Сортування у цьому місці необхідне для того, щоб виявити найбільш пристосовану хромосому, яка не повинна підлягати мутації.

9. *Мутація*. Використовуємо оператор мутації Uniform. Для цієї процедури задається відсоток ймовірності мутації генів у цілій популяції, окрім найкращої хромосоми. Загальну кількість генів у межах популяції, що підлягають мутації, визначають за формулою (17), згідно із заданим значенням відсотка мутації $Prob_{mut}$.

$$N_{genes\ mut} = \frac{Prob_{mut}}{100\%} \cdot N_{genes} \cdot (N_{chroms} - 1). \quad (17)$$

Для ідентифікації генів, що підлягають мутації, створюють двовимірний масив, перший індекс якого вказує на номер хромосоми в списку популяції, а другий індекс – на порядковий номер гена у вибраній хромосомі. Тобто спершу випадковим чином визначається номер хромосоми, а потім номер гена в цій хромосомі, і вибраний ген піддається мутації (за допомогою генератора випадкових чисел генерується нове нормалізоване значення для гена, яке не має жодної прив'язки до попереднього значення). Ця процедура повторюється $N_{genes\ mut}$ -разів.

10. *Обчислення фітнес-значення для нащадків, що зазнали мутації.* Обчислення фітнес-значення розпаралелюється залежно від кількості фізичних (логічних) ядер процесора персонального комп'ютера.

11. *Сортування популяції за фітнес-значенням.*

12. *Елітизм.* Ця процедура вставляє на початок популяції відібрані елітні хромосоми у п. 4 із заміною гірших хромосом.

13. *Сортування популяції за фітнес-значенням.* Цей крок можна пропустити, якщо значення елітизму дорівнює нулю.

14. *Створення копії покоління.* Ця копія буде необхідною, якщо ми вирішимо продовжити пошук оптимуму за допомогою генетичного алгоритму. Можлива ситуація, коли під час роботи генетичного алгоритму нам буде потрібно змінити деякі параметри як об'єкта оптимізації, так і самого алгоритму, наприклад, зменшити відсоток ймовірності мутації чи змінити межі пошуку для досліджуваних параметрів. Для цього нам потрібно буде зупинити роботу генетичного алгоритму, вибрати нові параметри та продовжити оптимізацію.

15. Якщо задану кількість поколінь пройдено, то завершуємо роботу генетичного алгоритму, а якщо ні, то переходимо до п.3.

Реалізували генетичний алгоритм мовою С#. Відповідно, ми скористалися вбудованим генератором псевдовипадкових чисел з рівномірним законом розподілу (клас Random) з бібліотеки .NET Framework, який побудований на основі субтрактивного алгоритму генератора випадкових чисел Д. Кнута. Для розпаралелення обчислень фітнес-значень використовували клас Task. Реалізацію генетичного алгоритму було апробовано на тестових задачах для оптимізації [10]: De Jong's function 1, Axis parallel hyper-ellipsoid function, Rotated hyper-ellipsoid function, Rastrigin's function 6, Schwefel's function 7. Для усіх тестових задач реалізований алгоритм знайшов визначені мінімуми та максимуми функцій.

Параметрична оптимізація системи керування автоматичним потенціометром

Розраховували динамічні режими роботи системи керування за таких значень. Максимальний кут повороту потенціометричних давачів $\theta_{\max} = 5$ рад; напруги насичення електронного підсилювача $U_{\max EP} = 15$ В та магнітного підсилювача $U_{\max MP} = 35$ В. Параметри мотора постійного струму: $L_A = 1,32$ Гн; $L_{FF} = 4,53$ Гн; $L_{AF} = L_{FA} = 0$ Гн; $r_A = 20,5$ Ом; $r_F = 67$ Ом; $c = 74,8$ Н·м/(Вб·А); $u_f = 30$ В, $k = 0,04$. Момент інерції ротора мотора $I_M = 0,01$ Н·м·с²/рад, момент інерції навантаження $I_H = 0,8$ Н·м·с²/рад, статичний об'єкт навантаження $M_{CH} = 1,04$ Н·м.

Досліджували оптимізацію системи керування за 5 параметрами, для яких було встановлено такі межі: опорна напруга $U_{on} = [0,02; 1]$ В, коефіцієнт підсилення електронного підсилювача $K_{EP} = [50; 1000]$, коефіцієнт підсилення магнітного підсилювача $K_{MP} = [1,1; 8]$, стала часу магнітного підсилювача $T_M = [0,001; 0,05]$ сек, коефіцієнт редукції $K_{ред} = [8; 42]$.

Для оптимізації динамічної характеристики системи керування ми використали описаний вище генетичний алгоритм. Як критерій для значення фітнес-функції ми вибрали відхилення дискретних миттєвих точок вихідної динамічної характеристики системи від задаючого значення. Тобто фітнес-значення розраховували за формулою:

$$Fitness = \sum_i |\theta_{ex} - \theta_{vux}(t_i)|. \quad (18)$$

З метою покращення ефективності від оптимізації фітнес-значення обчислювали для різних задавальних значень $\theta_{ex} = \{0,1; 3; 5\}$ радіан та у результаті додавалися до єдиного значення. Моделювали систему керування методом Рунге–Кутта четвертого порядку. Інтегрували систему на інтервалі часу 10 секунд.

При заданих параметрах за допомогою генетичного алгоритму було віднайдено оптимум для динамічної характеристики системи керування (рис. 4). За малих значень θ_{ex} виникає статична помилка в системі керування. Починаючи з $\theta_{ex} = 1,6$ рад статична помилка в системі практично щезає.

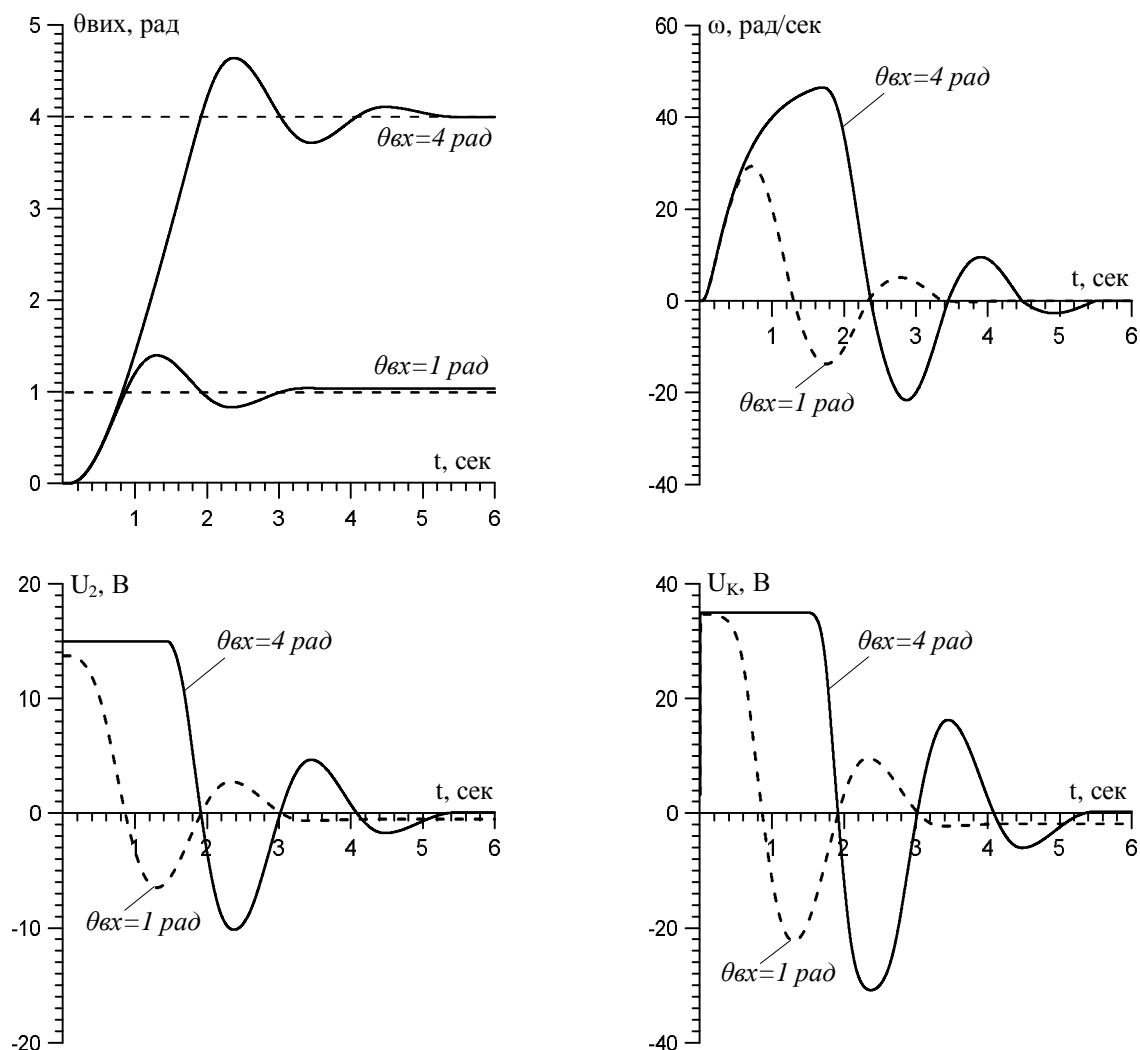


Рис. 4. Динамічні характеристики системи керування для оптимізованих параметрів:

$$U_{on} = 0,24 \text{ В}; K_{ЕП} = 340,1; K_{МП} = 3,49; T_m = 1,02 \cdot 10^{-3} \text{ сек}; K_{ред} = 15,8$$

Отримані внаслідок оптимізації динамічні характеристики системи керування (рис. 4) містять перерегулювання вихідної величини порядку 16 % від задавального значення. Тому ми вирішили відшукати оптимум, наклавши обмеження до 1 % перерегулювання вихідної характеристики. З цією метою для фітнес-функції накладається додаткова умова. Це посилює пошук оптимуму згідно із поставленою умовою.

$$\text{якщо } (\theta_{вих} > 1,01 \cdot \theta_{ex}) \text{ тоді } \theta_{вих} = 10 \cdot \theta_{вих}$$

$$Fitness = \sum_i |\theta_{ex} - \theta_{вих}(t_i)| \quad (19)$$

На рис. 5 наведено результат нашого пошуку. На відміну від попереднього результату, вихідна характеристика $\theta_{вих}$ наростає плавно до заданого значення.

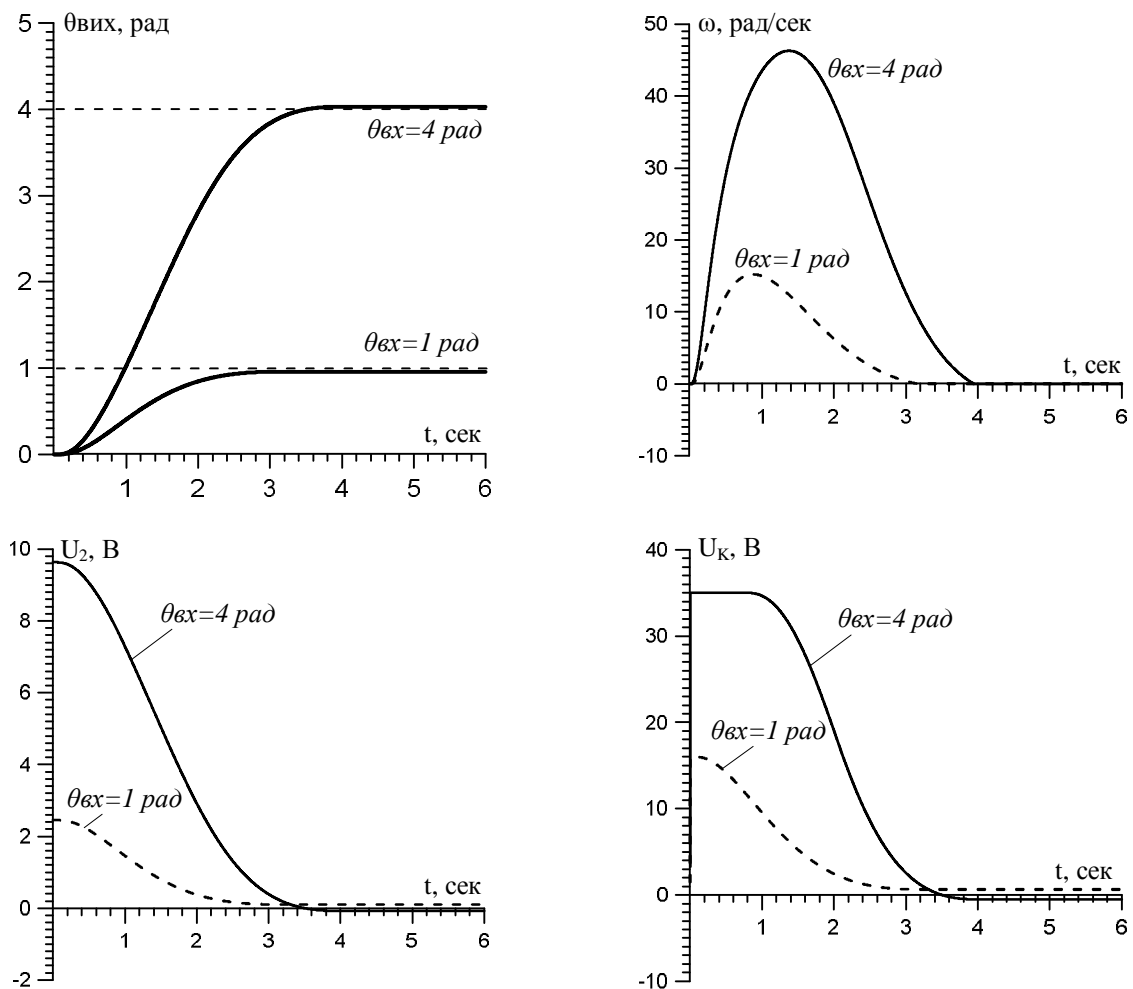


Рис. 5. Динамічні характеристики системи керування для оптимізованих параметрів:
 $U_{on} = 0,089 \text{ В}; K_{ЕП} = 137,8; K_{МП} = 6,53; T_M = 1,01 \cdot 10^{-3} \text{ сек}; K_{ред} = 24,9$

На рис. 6 наведено динамічну характеристику для останнього варіанта оптимізованих параметрів системи керування автоматичним потенціометром. Задавальні значення $\theta_{вх}$ змінювалися кожні 5 секунд (3,1 та 4,5 рад). При цьому система керування показала достатнього якісну динамічну характеристику та стійкий характер.

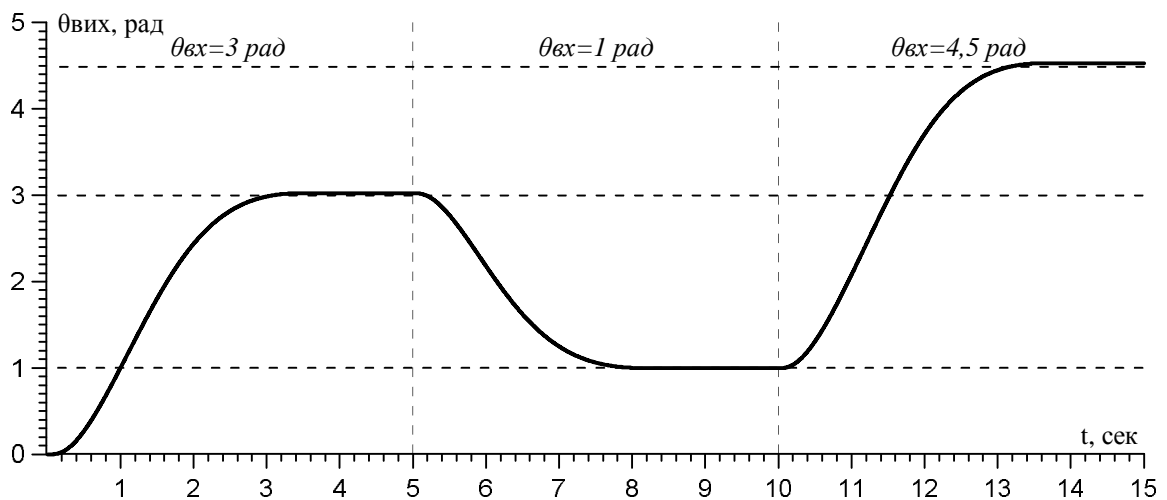


Рис. 6. Динамічна характеристики системи керування при зміні задавального значення

Висновок

За допомогою бінарного генетичного алгоритму було проведено параметричну оптимізацію системи керування автоматичним потенціометром. Оптимізація дала можливість отримати якісні динамічні характеристики системи керування. Розроблена математична модель системи керування є універсальною та простою для алгоритмізації.

1. Yann Collette, Patrick Siarry. *Multiobjective Optimization: Principles And Case Studies*. – Springer-Verlag Berlin Heidelberg. – 2004. – 293 p. 2. Reiner Horst and Tuy Hoang. *Global Optimization: Deterministic Approaches, 3rd edition*. – Springer-Verlag GmbH: Berlin, Germany. – 1996. – 696 p. 3. Mordecai Avriel. *Nonlinear Programming: Analysis and Methods*. – Dover Publications: Mineola, NY, USA. – 2003. 4. G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. MarchettiSpaccamela, M. Protrasi, *Complexity and Approximation Combinatorial Optimization Problems and their Approximability Properties*. – Springer, Berlin, Heidelberg. – 1999. – 524 p. 5. Thomas Weise. *Global optimization algorithms: theory and application, 3rd Edition*. – Thomas Weise – 2011. – 1217 p. 6. S.N. Sivanandam, S.N. Deepa. *Introduction to Genetic Algorithms*. – Springer-Verlag Berlin Heidelberg. – 2008. – 456 p. 7. Randy L. Haupt, Sue Ellen Haupt. *Practical genetic algorithms*. – 2nd ed. – John Wiley & Sons, Inc., Hoboken, New Jersey. – 2004. – 272 p. 8. Mitchell Melanie. *An Introduction to Genetic Algorithms*. – A Bradford Book The MIT Press. – 1999. – 162 p. 9. Субботін С. О., Олійник А. О., Олійник О. О. *Неітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей*. - Запоріжжя: ЗНТУ. – 2009. – 375 с. 10. Hartmut Pohlheim. *Examples of Objective Functions*. – www.geatbx.com – 2006. – 21 p. 11. Самотій В. В. *Математичне моделювання стаціонарних процесів електроматнетних пристроїв систем керування*. – Львів: Фенікс. – 1997. – 170 с.