ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ТА ГЕНЕТИЧНА ОПТИМІЗАЦІЯ СИСТЕМ КЕРУВАННЯ ЗАСОБАМИ ПРОГРАМУВАННЯ LABVIEW

В.А. Жеребко, О.А. Писаренко, В.П. Драбинко

Процедури проектування, синтезу, імітаційного моделювання та цільової реалізації алгоритмів вирішення оптимізаційних задач у ієрархічних автоматизованих системах керування складними об'єктами вимагає використання різноманітного математичного та програмно-апаратного забезпечення впродовж науково-дослідних та проектних робіт. У даній статті пропонується уніфікація математичного та алгоритмічного забезпечення у вигляді єдиної концепції вирішення задач оптимального керування. А у якості уніфікації програмно-апаратних засобів реалізації математично-алгоритмічних напрацювань пропонується використання програмного забезпечення LabVIEW та промислового контролера CompactRIO, на базі яких розроблено програмний інструментарій вирішення задач оптимізації із використання генетичного алгоритму.

Ключові слова: імітаційне моделювання, автоматизація систем керування, оптимізація систем керування, LabVIEW, генетичний алгоритм, генетичний оптимизатор.

Процедури проектування, синтезу, імітаційного моделювання та цільової реалізації алгоритмів рішення оптимізаційних задач в ієрархічних автоматизованих системах управління складними об'єктами потребує використання різноманітного математичного та програмно-апаратного забезпечення в течіні науково-дослідницьких та проектних робіт. В даній статті пропонується уніфікація математичного та програмно-апаратного забезпечення в вигляді єдиної концепції вирішення задач оптимального керування. А в якості уніфікації програмно-апаратних засобів реалізації математично-алгоритмічних нараців робіт пропонується використання програмного забезпечення LabVIEW та промислового контролера CompactRIO, на базі яких розроблено програмний інструментарій вирішення задач оптимізації по використанню генетичного алгоритму.

Ключові слова: імітаційне моделювання, автоматизація систем управління, оптимізація систем управління, LabVIEW, генетичний алгоритм, генетичний оптимизатор.

The procedures for the engineering, synthesis, simulation and targeted implementation of algorithms for solving optimization problems in hierarchical automated control systems require complex objects using a variety of mathematical and software and hardware for research and engineering work. In this article the unification of mathematical and algorithmic support in the form of a unified concept of solving optimal control is proposed. We propose to use LabVIEW software and CompactRIO industrial controller as the unification of software and hardware implementation of mathematical-algorithmic developments, on which developed software tools to solve optimization problems using genetic algorithm.

Key words: simulation, automation of control systems, optimization of control systems, LabVIEW, genetic algorithm, genetic optimizer.

Вступ

Вирішення оптимізаційних задач керування в складних системах не мають традиційного простого рішення і в кожному конкретному випадку потребують трудомістких наукових розробок [1]. Це пояснюється тим, що оптимізаційні задачі – вкрай складні процедури, що передбачають вивчення та імітаційне моделювання складних об’єктів керування (ОК), стратегій та ситуацій, створення альтернатив та пошуку варіантів рішень на альтернативах, виходячи з конкретного ситуаційного стану ОК та навколишнього середовища. Можливість реалізації таких процедур обмежується з однієї сторони реальною динамікою ОК і швидкодією технічних засобів реалізації ієрархічних автоматизованих систем керування (АСК), з іншої – різноплановістю, складністю та неефективністю алгоритмів вирішення таких задач, особливо в умовах невизначеності, зміни параметрів ОК, збурень, зміни критеріїв функціонування та інше. Все це призводить до того, що оптимізаційні задачі в АСК в повній мірі практично не реалізуються, хоч повинні вирішуватися на кожному рівні ієрархії АСК. При вирішенні оптимізаційних задач в ієрархічних АСК на всіх рівнях можна застосовувати, як основу, методику вирішення двоетапної задачі оптимального керування (ЗОК).

Суть єдиної концепції вирішення ЗОК полягає в її нумерованому підхіді до вирішення оптимізаційних задач на всіх рівнях ієрархії АСК. Відповідно до цієї концепції оптимізаційні задачі на всіх рівнях представляються узагальненою моделлю у вигляді кортежу, що об’єднує модель ОК, критерій, які представляють оптимізаційну задачу, обмеження на керуючі впливи і стани ОК та проміжок часу на якому вирішується оптимізаційна задача. Оптимізаційні задачі вирішуються шляхом генетичного алгоритму на моделях АСК, як задачі параметричної оптимізації регуляторів. Моделі ОК представляються лінгвістичними або нейроморфними структурами [2], параметри яких також налагоджуються генетичним алгоритмом [3], як задачі параметричної оптимізації. Такий підхід до вирішення оптимізаційних задач керування зводить весь процес до евристичної і дозволяє уникнути використання складних математичних процедур, як при побудові моделей ОК, так і при вирішенні оптимізаційних задач.

Едина концепція реалізує методику вирішення двоетапної задачі оптимізації на всіх рівнях ієрархії АСК у такій узагальненій послідовності:
- представити ЗОК яка включає в себе кортеж, що об’єднує модель ОК, критерій, які представляють оптимізаційну задачу, обмеження на керуючі впливи і стани ОК та проміжок часу на якому вирішується оптимізаційна задача;
- розробити модель ОК, виходячи із априорних та експериментальних даних;
© В.А. Жеребко, О.А. Писаренко, В.П. Драбинко, 2018
ISSN 1727-4907. Проблеми програмування. 2018. № 2–3. Спеціальний випуск
- виходячи з моделі ОК синтезувати структуру регулятора і побудувати модель системи керування;
- знайти оптимальні параметри налагодження регулятора, що вирішують оптимізаційну задачу на моделі системи керування;
- шляхом моделювання роботи моделі системи керування знайти множину оптимальних програм керування ОК для різних допустимих умов і станів його функціонування, створити необхідний банк оптимальних програм керування;
- реалізувати замкнуту систему керування ОК з оптимально налаштованим регулятором;
- для вирішення оптимізаційної задачі використовувати відповідну оптимальну програму керування із банку оптимальних програм та керування оптимального регулятора;
- одержану додаткову інформацію про якість вирішення оптимізаційної задачі в процесі функціонування системи використати для адаптації системи керування.

При вирішенні багатокритеріальних оптимізаційних задач використовуються технології скаляризації, які відносяться до нерівних величин, а також до інтервальних і інших типів; використання інтелектуальних технологій, зокрема нечітких систем, з різноманітними програмними засобами та технічними пристроями [4]. Такі розроблені одиниці вирішують оптимізаційні задачі за допомогою інтелектуальних систем керування та технології її реалізації. Технологія реалізації, її обліку потребує програмних середовищ для створення і моделювання лінгвістичних та нейромережевих систем керування та програмних засобів для реалізації оптимізаційних процедур генетичного пошуку оптимальних параметрів моделей об'єктів і регуляторів.

Математичне вирішення оптимізаційних задач це достатньо довготривалий та витратний обчислювальний процес. Зазвичай вирішення таких задач відбувається на етапі дослідження ОК, проектування АСК, імітаційного моделювання системи в цілому, до початку режиму експлуатації АСК на реальному ОК. Сучасний стан автоматизації ОК та побудови АСК вимагає від систем здатності до вирішення ЗОУ реальному часу. Часто густо процедури моделювання, проектування, програмування, цільового викопання виконуються на зовсім різних програмно-апаратних платформах, різноманітними програмними засобами та технічними пристроями [4]. Тож окрім уніфікації математичного апарату щодо вирішення ЗОУ в складних АСК постає також питання у використанні моделювання та програмного засобу для реалізації генетичного алгоритму (ГА). Це потребує здатності системи до вирішення ЗОУ з допомогою генетичного алгоритму (ГА) в складних АСК постає також питання у використанні моделювання та програмного засобу для реалізації генетичного алгоритму (ГА). Це потребує здатності системи до вирішення ЗОУ з допомогою генетичного алгоритму (ГА)

У даній статті наведений узагальнений опис програмного інструментарію що націлений на вирішення пошукових задач оптимізації, зокрема налаштування оптимального регулятора. Загальна структурна система GenOp наведена на рис. 1. Перелік основних функцій інструментальної системи GenOp:

1. Програмний засіб для моделювання й оптимізації систем керування

Програмний засіб (ПЗ) для оптимізації систем керування реалізовано у вигляді бібліотеки модулів "Генетичний оптимізатор" (GenOp) та з багатьох віртуальних програмних інструментів VI (програм і даних), які розширюють бібліотеку стандартних програмних модулів та засобів автоматизованого проектування, що реалізовані виробником в LabVIEW. Базова задача GenOp – це відображати стратегію оптимального керування технічного компоненту, який виконує оптимальний алгоритм (ГА) на реальному часу.

1. Програмний засіб для моделювання й оптимізації систем керування

Програмний засіб (ПЗ) для оптимізації систем керування реалізовано у вигляді бібліотеки модулів "Генетичний оптимізатор" (GenOp) та з багатьох віртуальних програмних інструментів VI (програм і даних), які розширюють бібліотеку стандартних програмних модулів та засобів автоматизованого проектування, що реалізовані виробником в LabVIEW. Базова задача GenOp – це відображати стратегію оптимального керування технічного компоненту, який виконує оптимальний алгоритм (ГА) на реальному часу. Програмний засіб включає у себе як базу програмних засобів, так і власну бібліотеку модулів та засобів автоматизованого проектування.
Прикладне програмне забезпечення

оптимізації (ЦСО);
- вибір КО та створення власних критеріїв користувача;
- налаштування параметрів оптимізації керування ОК.

Цільові структури оптимізації (ЦСО) – у загальному сенсі це закінчені структурні елементи системи GenOp, що складаються з математичних описів та потребують пошуку власних оптимальних параметрів за певним критерієм, наприклад: глобальний мінімум функції багатьох змінних, навчання штучної нейромережі зі невідомим вектором ваг, побудова нечіткої моделі зі невідомими правилами та функціями принаймні до тощо. Тобто ЦСО – це така математична структура, що обов’язково містить невідомий вектор (кластер, масив) параметрів, пошук яких виконує інтелектуальний оптимізаційний ГА. Оскільки системи GenOp функції оптимізаційного пошуку покладаються на ГА, тому в термінах еволюційного апарату ЦСО – це не що інше як фітнес-функція прилаштованості індивідів.

Система GenOp спроектована на базі програмної платформи LabVIEW із використання графічної мови програмування G та допоміжного модулю Control Design & Simulation Toolkit (CDS). LabVIEW – повноцінний інструментальний пакет із широким математичним апаратом та редактором візуальних операторських панелей. Модуль CDS розширює стандартний набір функцій LabVIEW для проектування та моделювання автоматичних систем керування. GenOp складається із наступних головних частин (рис. 1): інтерфейсна підсистема (лицьова панель) та підсистема реалізації (блок-діаграми). Ліцьова панель GenOp дозволяє вводити наступні налаштування: початкові дані моделі ОК та її складових елементів, обирати алгоритм, структуру й параметри оптимізації (це ЦСO), обирати робочий критерій оптимізації, а також відображати результати обрахунків у вигляді таблиць та графіків. Блок-діаграми підсистеми реалізації містять всі відкриті вихідні коди системи GenOp, а саме: модуль генетичної оптимізації, модуль критеріїв оптимізації та набір шаблонів із прикладами ЦСО. Відкритість вихідних кодів всіх блок-діаграм не обмежує функцій відлагодження та подальшого вдосконалення й модернізації GenOp. Зазвичай будь-яке ПЗ поділяється на окремі функціонально закінчені модулі, кожен з яких виконує окремий блок завдань та оперує лише певним набором даних, визначеного типу та об’єму для збереження у оперативній або постійній пам’яті. Модулі інтерфейсу та введення даних відокремлюються в окрему групу. Система GenOp має у складі наступні модулі:
- модуль «Генетична оптимізація»;
- модуль «Обрахунок критеріїв оптимізації»;
- модулі «Цільова структура оптимізації».

![Рис. 1. Структура системи GenOp](image)

Інтерфейс системи GenOp надає користувачу наступні можливості:
- введення початкових даних та налаштувань, необхідних для роботи системи;
- спостерігання за динамікою оптимізаційних обрахунків на цифрових індикаторах та віртуальному осциллографі;
- запуск системи на виконання та його зупинка.

Програмний модуль «Генетична оптимізація» (ГО) спроектовано із використанням відкритого некомерційного допоміжного модулю WAPTIA, написаний в середовищі LabVIEW мовою G. Бібліотека WAPTIA містить базові елементи реалізації ГА, необхідні для повноцінного функціонування системи GenOp. За замовчування WAPTIA виконує оптимізаційний пошук глобального максимуму фітнес-функції, а GenOp розроблено для оптимізаційного пошуку глобального мінімуму.

Програмний модуль «Цільова структура оптимізації» (ЦСО) виконує обрахунок математичної моделі регулятора та моделі ОУ об'єднаних у контур керування за умови наявності вхідних та вихідних даних. Програмний модуль «Критерій оптимізації» (КО) обраховує оптимізаційний алгоритм пошуку оптимуму (глобальний та/або локальні, мінімум чи максимум) враховуючи початкові дані про стан ОУ та граничні обмеження.

Програмний модуль «Генетична оптимізація» складається із наступних частин: лицьова панель модулю та блок-діаграми ГА. Лицьова панель дозволяє вводити налаштування параметрів ГА, вибір режимів роботи ГА, а також відображати результати оптимізаційних обрахунків у вигляді таблиць та графіків. Блок-діаграми підсистеми реалізації містять всі відкриті вихідні коди програмного модулю «Генетична оптимізація», а саме: модуль ініціалізації індивідів, модуль обчислення фітнес-функції ЦСО, модуль селекції та схрещування та також модуль мутації індивідів. Відкритість модуль ініціалізації індивідів, модуль мутації індивідів, модуль селекції та модуль схрещування необхідні для повноцінного функціонування системи GenOp та можуть бути додані в будь-який момент без настання поточного режиму. Модулі селекції та схрещування можуть бути відключено і відключені за рішенням користувача.

До модулю ГО перед початком роботи системи GenOp підключається тільки один із модулів ЦСО, а модулі KО можуть бути використані для керування старим модулем ОУ ГО. Відкритість інтерфейсу системи GenOp може бути додано і використано для керування старим модулем ОУ ГО.

Узагальнений алгоритм роботи системи GenOp складається з кроків.

**Крок 1.** Пошук оптимальних точок. Серед реалізованих програмного модулю KО обираються необхідні блок-діаграми та формується програмний фрагмент, який надалі використовується для виконання налаштування параметрів ГА, вибору режимів роботи ГА, а також відображати результати оптимізаційних обрахунків у вигляді таблиць та графіків. Блок-діаграми підсистеми реалізації містять всі відкриті вихідні коди програмного модулю «Генетична оптимізація», а саме: модуль ініціалізації індивідів, модуль обчислення фітнес-функції ЦСО, модуль селекції та схрещування та також модуль мутації індивідів. Відкритість вихідних кодів необхідних блок-діаграм не обмежує функції відкладання та подальшого вдосконалення модуля програмного модулю «Генетична оптимізація».

**Крок 2.** Обирається режим роботи системи GenOp, що здатна функціонувати у двох режимах: режим моделювання (статичний) та режим реального часу (динамічний). У випадку вибору режиму реального часу – перехід до кроку 6, інакше – крок 3.

**Крок 3.** Вибір ЦСО, що складається із регулятора та моделі ОУ. Вибір виконується шляхом введення шляху та імені файлу віртуального інструмента із відповідним ЦСО.

**Крок 4.** Виконання модулю ГО. Перед початком роботи вводяться необхідні налаштування параметрів ГА, вибору режимів роботи ГА, а також відображати результати оптимізаційних обрахунків у вигляді таблиць та графіків. Блок-діаграми підсистеми реалізації містять всі відкриті вихідні коди програмного модулю «Генетична оптимізація», а саме: модуль ініціалізації індивідів, модуль обчислення фітнес-функції ЦСО, модуль селекції та схрещування та також модуль мутації індивідів. Відкритість вихідних кодів необхідних блок-діаграм не обмежує функції відкладання та подальшого вдосконалення модуля програмного модулю «Генетична оптимізація».
вирішуються практично всі типи завдань оптимізаційного характеру.

Керуючі параметри модулю ГО:
- режим мутації: кожен біт інвертується з певною ймовірністю; інвертуються ті біти, що збігаються з батьківськими хромосомами (координатоми) з певною ймовірністю; мутація відсутня;
- обсяг буфера «розгону» для розміщення вихідної популяції;
- обсяги наступних популяцій (масивів);
- число ітерацій (поколінь, кроків оптимізації).

Робота ГА у модулі ГО полягає у наступному. Перед першим кроком потрібно випадковим чином створити початкову популяцію індивідів; наявні якщо вона виявиться абсолютно неконкурентоздатною, ймовірно, що ГА все одно достатньо швидко переведе її в життєздатну популяцію. Тож підсумком першого кроку алгоритму є популяція \( H \), що складається з \( N \) особин. Розмноження в ГА зазвичай статеве – щоб створити нащадка (біологічний принцип), потрібні деяка кількість батьків, зазвичай двоє. Розмноження в різних ГА визначається по різному: з індивідуалізації передатної функції, матрицею простору станів \( \text{м} \), що ми вже обговорюємо в попередніх лекціях. ГА використовує алгоритм відшуковує оптимальні параметри ПІД, що використано ще для класичного ПІД.

Алгоритм відшуковує оптимальні параметри ПІД:
- використовуючи логарифмичний критерій, що визначається в режимі тестування, вівтарюється більш міцна комбінація параметрів, що відповідає кращій компромісній відповіді до задачі; для того, аби створити нащадка, було використано дві типи ЦСО, а саме: на прикладі АСК із модулем ОФК коливального типу, а також в ЦСО нейроморфного типу, що використано для оптимізації параметрів КОН в системі GenOp, яка є критерієм мінімізації енерговитрат на усіх рівнях ієрархії АСК.

Цільові структури оптимізації (ЦСО) – це завершені віртуальні інструменти (VI-файли), розроблені в LabVIEW, що містять у собі моделі замкнутих контурів керування. ЦСО складаються здебільшого із блоку моделі OK та регулятора для різного рівня складності моделей OK та законів керування ними.

Можливості пакету LabVIEW дозволяють побудувати наступні види ЦСО як на моделях ОК так і на реальних об'єктах. Це досягається завдяки інтеграції у LabVIEW програмних засобів підтримки апаратного забезпечення (модулі введення–виходу) головного контролера CompactRIO із засобами вимірювання, регулювання та керування, встановлених безпосередньо на цих ОК.

Наведемо стислій перелік окремих елементів ЦСО, підтримка якіх реалізована в системі GenOp:
- модель OK та регулятор визначені аналітично та задані у формах: диференційного рівняння (системи) із початковими та граничними умовами, у вигляді передатної функції, матрицею простору станів (Control Design Toolkit);
- модель OK та регулятор визначені аналітично та побудовані емпіричним способом різними методами ідентифікації (System Identification);
- модель OK та регулятор побудовані в формі нейросетевих моделей штучного інтелекту (Fuzzy Logic);
- модель OK та регулятор побудовані інтелектуальними методами нейроморфного апарату (Neural Net);
- модель OK та регулятор визначені аналітично та побудовані емпіричним способом різними методами ідентифікації (System Identification);
- модель OK та регулятор побудовані в формі нейросетевих моделей штучного інтелекту (Fuzzy Logic);
- модель OK та регулятор побудовані інтелектуальними методами нейроморфного апарату (Neural Net).
2. Програмна реалізація генетичного оптимізатора

Відповідно до розробленого алгоритму функціонування ГО побудовано віртуальний інструмент ГО (рис. 2), що отримує із лицьової панелі системи GenOp початкові параметри налаштувань та шлях до файлу із ЦСО у вигляді структури об'єднаних даних. Розділені окремо кластерні дані подаються до блоку ГА.

Початкові масиви розв'язку та популяцій обнулено. Модуль ГО також містить цикл для оновлення даних на графіку, що відображає процес оптимізаційного пошуку за допомогою ГА. ГО здебільшого базується на підпрограмах віртуальних інструментів реалізації ГА. Відповідно до класичної теорії та алгоритму роботи ГА розроблено наступні модулі ГА блоку ГО для системи GenOp.

Модуль «Ініціалізація популяцій» реалізує створення (генерування) початкового набору даних про популяцію індивідів для подальшого генетичного пошуку в роботі ГА. Розмірність масиву задається на лицьовій панелі системи GenOp, закладка «Parameters».

Модуль «Сортування популяцій» виконує процедуру сортування поточних популяцій індивідів відповідно до принципу природного відбору, за яким найбільші шанси на участь у створенні нових особин мають хромосоми з найбільшими значеннями функції пристосованості. Існують різні методи селекції. В системі GenOp використовується метод рулетки (roulette wheel selection), який своїй назві отримав за аналогією з відомою азартною грою. Кожній хромосомі може бути співставленний сектор колеса рулетки, величина якого встановлюється пропорційно значенням функції пристосованості даної хромосоми. Тому, чим більше значення функції пристосованості, тим більше сектор на колесі рулетки.

Модуль «Кодування хромосом фенотипу у генотип» призначений для перетворення двійкових записів хромосом до еквівалентних числових значень. У класичному ГА застосовуються два основних генетичних оператори: оператор схрещування (crossover) і оператор мутації (mutation). Функцію crossover виконує модуль «Схрещення батьківських хромосом», а mutation – модуль «Мутації». Однак слід зазначити, що оператор мутації грає явно другорядну роль порівняно з оператором схрещування. Це означає, що схрещування в класичному ГА здійснюється практично завжди, тоді як мутація – досить рідко. Ймовірність схрещування, як правило, досить велика (найчастіше), тоді як імовірність мутації встановлюється дуже малою (найчастіше). Це впливає на аналогії зі світом живих організмів, де мутації відбуваються надзвичайно рідко.

Рис. 2. Блок-діаграма віртуального інструменту генетичного оптимізатора
У ГА мутації хромосом може виконуватися на популяції батьків перед схрещуванням або на популяції нащадків, утворених в результаті схрещування.

На першому етапі схрещування вибираються пари хромосом з батьківської популяції (батьківського пулу). Це тимчасова популяція, що складається з хромосом, відібраних в результаті селекції i призначених для подальших перетворень операторами схрещування і мутації з метою формування нової популяції нащадків.

На даному етапі хромосоми з батьківської популяції об’єднуються в пари. Це проводиться випадковим чином або відповідно до ймовірності схрещування. Далі для кожного пари відібраних таким чином батьків розігрується позиція гена (локус) у хромосомі, що визначає точку схрещування. У результаті схрещування пари батьківських хромосом виходить наступна пара нащадків. Модуль «Декодування хромосом генотипу у фенотипу» призначений для зворотного перетворення еквівалентних числових значень хромосом до їх двійкових записів (бінарний вигляд).

Модуль ГО у статичному режимі використовується для пошуку оптимальних керувань, оптимальних параметрів моделей ОК, регуляторів тощо без урахування реальної поведінки реального ОК в реальному часі. Навіть отримуючи оптимальний набір параметрів керування та налаштування регулятора не існує стовідсоткової гарантії, що в результаті оптимального керування щось не вийде. Це звичайно пов’язано з тим, що з плином часу ОК перебуває у певному життєвому циклі функціонування, в осередку якого оптимізаційний пошук та власне синтез оптимального керування взагалі не виконується. Ще складніша ситуація виникає коли ОК априорі є слабкоформалізованим і не ідентифіковане з певними властивостями невизначеного характеру. Тому необхідність виконувати ГО у реальному часі в системах керування ОК є досить актуальною та навіть необхідною проблемою.

В системі GenOp функціональний, що реалізує режим роботи ГО в реальному масштабі часу реализованый на базі стекової структури FIFO, яка зберігає фрагмент життєдіяльності ОК в реальному масштабі часу. Система GenOp відкрита розробнику для подальшого вдосконалення та розширення стеку кадрів (кадр) за певний попередній проміжок часу. Глибина стеку – змінний параметр та налаштується окремо. В процесі функціонування системи керування, у разі виходу стану ОК за межі оптимальних керувань, вмикається обережне керування кадру ОК та перерахунок оптимальних параметрів ЦСО засобами модулю ГО.

Висновки

Розроблені концепція і програмний інструментарій засобами LabVIEW для реалізації та методики синтезу оптимальних програмних керувань GenOp дозволяють довести складність побудови систем керування, що вирішують оптимізаційні задачі в складних реаліях однооб’єктних систем. Под ред. Н.Д. Егупова. М.: Изд-во МГТУ им. Н.Э. Баумана, 2000. 748 с.

Література

1. Методы классической и современной теории автоматического управления: Учебник в 3-х т. Т.2: Оптимизация многообъектных многокритериальных систем. Под ред. Н.Д. Егупова. М.: Изд-во МГТУ им. Н.Э. Баумана, 2000. 748 с.
2. Хайдин С. Нейронные сети: полный курс.: 2-е издание (пер. с англ.). М.: Издательский дом «Вильямс», 2006. 1104 с.
3. Курейчик В.М. Генетические алгоритмы и их применение // Publishing house of TRTU, second edition, 2002. 242 с.
4. Жеребко В.А, Кравець П.І., Василевська Х.С., Степанчук О.О. Підхід до побудови штучних нейронних мереж та їх реалізації на базі стекової структури FIFO, яка зберігає фрагмент життєдіяльності ОК. (кадр) за певний попередній проміжок часу. Глибина стеку – змінний параметр та налаштується окремо. В процесі функціонування системи керування, у разі виходу стану ОК за межі оптимальних керувань, вмикається обережне керування кадру ОК та перерахунок оптимальних параметрів ЦСО засобами модулю ГО.

Висновки

Розроблені концепція і програмний інструментарій засобами LabVIEW для реалізації та методики синтезу оптимальних програмних керувань GenOp дозволяють довести складність побудови систем керування, що вирішують оптимізаційні задачі в складних реаліях однооб’єктних систем. Под ред. Н.Д. Егупова. М.: Изд-во МГТУ им. Н.Э. Баумана, 2000. 748 с.

Література

1. Methods of classical and modern theory of automatic control: A textbook in 3 volumes. T.2: Optimization of multi-object multicriteria systems. Egupov N.D. // Moscow: MSTU them. N.E. Bauman, 2000. 748 p. (in Russian)
2. Haykin S. Neural Networks: A Comprehensive Foundation. Moscow: 2006. 1104 p. (in Russian)
3. Kureichik V.М. Genetic algorithms and their application // Publishing house of TRTU, second edition, 2002. 242 p. (in Russian)
4. Zherebko V.A., Kravets P.I., Vasilevsky H.C., Stepanchuk O.O. An approach to the construction of artificial neural networks algorithms for solving optimization problems in real-time control systems. // Computing Intellect, 2011: ChDTU, 2011. P. 165. (in Ukrainian)
5. Zherebko V.A., Onatsky A.V., Solomonuyk Yu.R. Investigation of the cost of FPGA resources when constructing the neural network logic function of XOR using LabVIEW // Computing Intellec t, 2011: ChD TU, 2011. P. 165. (in Ukrainian)
6. Zherebko V.A. Software model of the genetic optimizer of artificial neural network weights in LabVIEW. Problems in programming. (2). P. 100–108. (in Ukrainian).
Про авторів:

Жеребко Валерій Анатолійович, старший викладач кафедри автоматики і управління в технічних системах Нціонального технічного університету України “КПІ імені Ігоря Сікорського”. Кількість наукових публікацій в українських виданнях – понад 20. Кількість наукових публікацій в зарубіжних виданнях – 2. http://orcid.org/0000-0001-6163-0258.

Писаренко Олег Анатолійович, студент кафедри автоматики і управління в технічних системах Нціонального технічного університету України “КПІ імені Ігоря Сікорського”.

Драбинко Віталій Павлович, студент кафедри автоматики і управління в технічних системах Нціонального технічного університету України “КПІ імені Ігоря Сікорського”.

Місце роботи авторів:

Національний технічний університет України “КПІ імені Ігоря Сікорського”. 03056, м. Київ-56, проспект Перемоги, 37, корпус 18. Тел.: +38 (044) 236 7989. E-mail: zherebko@kpi.ua, oa.pisarenko@gmail.com, v.drabynko@gmail.com.