

УПРАВЛІННЯ ПРОЕКТАМИ ТА ПРОГРАМАМИ

УДК 004.73

Н. І. Бойко, В. Ю. Михайлишин
Національний університет “Львівська політехніка”,
кафедра інформаційних систем та мереж

ЕФЕКТИВНІСТЬ ЗАСТОСУВАННЯ ГЕНЕТИЧНИХ АЛГОРИТМІВ ДЛЯ ПОШУКУ ОПТИМІЗОВАНИХ РІШЕНЬ

© Бойко Н. І., Михайлишин В. Ю., 2016

Обґрунтовано теоретичні положення, запропоновано методичні та практичні рекомендації, що дають змогу підвищити дієвість функціонування інформаційної системи. Проаналізовано основні моделі генетичних алгоритмів для знаходження кращого рішення в інформаційній системі. Описано основні аспекти використання генетичних алгоритмів для застосування їх у еволюційному моделюванні. Представлено генетичний алгоритм для оперування критеріями вибору і відбору необхідної популяції рішень. Наведено існуючі підходи і методи застосування генетичних алгоритмів для розв’язання задач багатокритеріальної оптимізації. Проаналізовано обчислювальну складність генетичного алгоритму пошуку оптимальних рішень.

Ключові слова: інформація, інформаційна система, моделювання, генетичний алгоритм, штучний інтелект, еволюційне обчислення, хромосома, спадковість, мутація.

In the article the theoretical principles were justified, methodological and practical recommendations to enhance the effectiveness of the information system were proposed. The analysis of the basic models of genetic algorithms to find the best solutions in the information system was suggested. The basic aspects of genetic algorithms for their use in evolutionary modeling were described. The genetic algorithm for handling the selection criteria and selection decisions required population was suggested. The existing approaches and methods of using genetic algorithms for solving multi-objective optimization are provided. The computational complexity of genetic algorithm search for optimal solutions were analysed.

Keywords: system, information flow, information processes, “cloud computing”, corporate information systems, information technology.

Вступ. Загальна постановка проблеми

Найвагомішою характеристикою штучного інтелекту як галузі науки є широка область застосувань. Ця властивість дає змогу залучати цікаві події та їх закономірності з інших галузей знань, використовувати їх для майбутніх застосувань для вирішення певних проблем та завдань. Адаже багато теорій розглядали крізь призму штучного інтелекту. На думку багатьох науковців [6–9], відому всім теорію Ч. Дарвіна (1859 р.) вважають початком розвитку нового наукового напрямку досліджень – еволюційного моделювання.

Еволюційне моделювання (ЕМ) можна визначити як напрямок дослідження штучного інтелекту, в основу якого покладено запозичені з популяційної генетики принципи. Вони об’єднують комп’ютерні методи (генетичні алгоритми, генетичне програмування, еволюційне програмування та еволюційні стратегії) моделювання природних еволюційних процесів. Еволюційні перетворення в природних системах спрямовані на досягнення аналогічної мети, такої як в штучних системах при розв’язанні оптимізаційних задач. Тільки в останньому випадку критерієм

“пристосованості” розв’язання є його якість, оптимальність щодо інших досліджених точок пошукового простору тощо. Запозичення решти ключових властивостей еволюції – спадковості, мінливості та природного добору – виділило ЕМ в окрему методологію розв’язання оптимізаційних задач. Зокрема і таких, які неможливо розв’язати точними математичними методами через неповноту вхідних та вихідних даних, наявність обмежень й високої розмірності, зовнішніх умов, які динамічно змінюються та функціонують у конкурентному середовищі.

Загальною ідеєю ЕМ було замінити сам процес моделювання складної системи на моделювання її еволюції. В процесі цього використовували механізми природного розвитку в інтеграції зі складними системами опрацювання інформації. Так Дарвін сформулював основні закони розвитку органічного світу: спадковість, мінливість та природний відбір, які дають змогу ефективно організувати роботу складних інформаційних систем (ІС).

Аналіз останніх досліджень та публікацій

Основоположниками розвитку еволюційного моделювання як самостійного напрямку в галузі штучного інтелекту вважають роботи Дж. Холланда, Л. Фогеля, А. Овена, М. Уолша, І. Букатової, Л. Расстригіної та інших дослідників. Їхні праці [1–5] пов’язані зі спробами відтворити за допомогою математичних моделей і алгоритмів основні механізми еволюційних перетворень у природних системах. Вони характеризуються постійним досягненням оптимальних форм існування у поєднанні з мінімальними витратами ресурсів. У результаті було створено теорію еволюційного моделювання (еволюційних обчислень), яка стала фундаментом для розробки принципово нових комп’ютерних методів оптимізації, зокрема завдань, які до цього вважалися нерозв’язними. Опис та дослідження наведених методів визначають мету нашого дослідження та виділяють основні завдання наукового пошуку.

Особливості еволюційного моделювання в системах

Важко назвати будь-яку природну чи штучну систему, якій би не були властиві еволюційні процеси розвитку. Основним процесом є стратегія “виживає найсильніший”. Із застосуванням методів ЕМ для дослідження таких систем можна найточніше, не вдаючись до модифікації їхніх математичних моделей, визначити оптимальні характеристики їх функціонування, нові структури і форми їх розвитку (еволюції). При цьому використовують головну концепцію ЕМ штучних систем – замість моделювання системи у вже готовому вигляді слід створювати прості об’єкти та змушувати їх пройти крізь усі етапи еволюції до готової моделі системи. Тобто спостерігається їх самоорганізація в напрямку підвищення якості властивостей всієї системи [7, 9]. Практичні можливості еволюційного моделювання насамперед розглядають за головним методом еволюційних обчислень – генетичним алгоритмом. В ньому якнайповніше реалізовано механізми природних еволюційних процесів, зокрема селекція, відтворення і наслідування. При цьому гнучкі засоби регулювання інтенсивності цих процесів дають змогу керувати процесом пошуку рішень, гарантуючи визначення найкращого з них.

Еволюційні алгоритми відрізняються від інших числових методів особливою стратегією пошуку оптимального рішення моделюванням розвитку деякої ситуації замість безпосереднього обчислення відповіді з детермінованих формальних залежностей. Отож, застосування генетичного алгоритму дає перевагу в невизначених ситуаціях, коли існує декілька доволі вдалих, хоча і неочевидних рішень, при цьому інші методи пошуку рішень виявляються непридатними або малоефективними. Крім того, такі алгоритми ефективно формують шаблони адаптивної поведінки системи, оскільки хромосоми, які змогли “вижити” в процесі еволюції рішення, зберігають, по суті, колективний досвід багатьох поколінь. Тобто, еволюційне моделювання можна ефективно використовувати в таких випадках [8, 9]:

- застосування еволюційних алгоритмів для вивчення і модифікації окремих процесів природної еволюції;
- покращення наявних інформаційних систем наданням їм властивостей адаптивної поведінки і можливості самоорганізовуватись на основі засобів еволюційного моделювання;
- автоматизація вирішення різноманітних оптимізаційних завдань у науково-технічних галузях.

Основні ідеї та механізми еволюційного моделювання

Основне завдання підходу, названого еволюційним моделюванням, – повністю замінити створення моделі великого та складного об'єкта на створення штучної моделі його еволюції. Такий підхід розрахований на використання природних засобів еволюції при побудові великих і складних інформаційних систем для опрацювання та аналізу даних. У своїй теорії походження видів Ч. Дарвін розкрив і пояснив головний принцип розвитку органічного світу, описавши його через взаємну роботу таких трьох факторів: спадкова мінливість, боротьба за існування та природний відбір.

Теорія Дарвіна знайшла своє підтвердження та розвиток у генетиці та інших наукових сферах. Дослідники [10–13] визначили певні необхідні та достатні умови, що визначають неминучість еволюційного процесу в системах:

- спадкова мінливість, тобто мутація як передумова еволюції, її матеріал;
- боротьба за існування як контролювальний і спрямовувальний фактор;
- природний відбір як перетворювальний фактор.

Еволюційні обчислення – поняття, яке застосовується для загальної характеристики пошукових, оптимізаційних або навчальних алгоритмів, оснований на формальних принципах природного процесу еволюції. На відміну від точних способів математичного програмування, методи еволюції дають змогу знайти розв'язки, які є дуже наближеними до оптимальних, за цілком допустимий проміжок часу. Вона характеризуються значно меншим ступенем залежності від певних унікальних властивостей додатків (які є універсальними) і у результаті дають змогу отримувати кращу ступінь наближення до оптимального рішення, аніж загальновідомі евристичні методи. Основною властивістю цих методів для оптимізації обчислень в інформаційних системах є можливість розв'язання багатомодальних (які містять кілька локальних екстремумів) завдань з відносно великим ступенем розмірності. Отримання результатів стає можливим внаслідок об'єднання детермінованості та певного елемента випадковості.

Детермінованість таких механізмів – це моделювання процесів відбору, розмноження і успадкування, які контролюються визначеними вимогами та правилами, для яких діє еволюційний закон – “виживає сильніший”. Ще однією важливою складовою ефективності подібних обчислень є створення моделей процесів успадкування та розмноження.

Паралельно з еволюційними методами використовують модель мутаційного процесу як випадковий елемент у обчисленнях. Завдяки даній властивості певні рішення можуть випадково змінюватись, що, своєю чергою, відкриває новий напрямок для еволюції рішення і відповідно може зменшити час на пошук оптимального рішення в інформаційній системі.

Переваги і недоліки еволюційних обчислень

Автори зазначають такі переваги та недоліки еволюційних досліджень. До переваг належать:

- широка область для застосування обчислень у інформаційній системі;
- можливість проблемно-орієнтованого підходу до розв'язання задач підбору стартової популяції, комбінування обчислень методами еволюції з нееволоційними типами алгоритмів, в яких процес еволюції продовжується доти, поки в наявності є потрібні йому ресурси;
- можливість пошуку складних рішень за великої розмірності системи;
- немає обмеження до вибору виду функції мети;
- зрозумілість базових положень і схем еволюційних методів обчислення.

Недоліками еволюційних обчислень є:

- розрахунки, які не гарантують отримання оптимального рішення, а лише наближаються до нього;
- висока витрата ресурсів при обчисленнях, які можна зменшити за рахунок розпаралелення задач;
- низька ефективність на фінальних стадіях проходження алгоритму;
- невирішеність завдання автономного пристосування.

Основні поняття і базова схема генетичного алгоритму

Основним і найважливішим методом еволюційного моделювання є генетичні методи. Вони дають змогу розвинути модель, керуючись еволюційними властивостями. Генетичні алгоритми (ГА) полягають в моделюванні природної еволюції в замкнутій системі. Кінцевою метою даного алгоритму є пошук оптимізованого розв'язку складної задачі, в якій звичайні математичні моделі мають складну структуру, і застосування класичних способів їх вирішення є неможливим або надзвичайно ресурсозатратним та громіздким [5].

Умовою оптимізації за використання ГА є пошук найкращих можливих рішень або пошук розв'язків за одним чи кількома заданими критеріями. Для реалізації генетичного алгоритму необхідно вибрати структуру даних, яка визначатиме представлені рішення.

Властивість об'єктів представляється у вигляді набору певних параметрів і характеристик та поєднуватися в один запис в еволюції має назву хромосоми. Імітація генетики – це вибір батьківських особин певних хромосом, які є членами популяції; їхнє схрещування; відбір особин-нащадків для створення нових поколінь на основі оцінювання пристосованості і відповідності очікуваним результатам згідно з цільовою функцією. Перераховані властивості призводять до еволюційного покращення розв'язку поставленої задачі в інформаційній системі.

В ІС хромосома – це рядок бітів. Але це не означає, що ГА обмежені подачею інформації лише у вигляді бітів. У деяких рішеннях використовують цілочислове або рядкове подання даних. Попри те, що для більшості задач, які виникають, рядки зі змінною довжиною підходять краще. Наразі інформаційні структури з фіксованою довжиною даних найбільш поширені в дослідженнях певних науковців [13–15]. Будь-яка хромосома є об'єднанням декількох окремих підкомпонентів, які зветься генами. Кожен з них розташовується у різних позиціях (локусах) хромосоми та може зберігати у собі певні значення – алелі. Якщо проводити паралель з бінарним рядком, то ген відповідає біту, локус – інформації про його позицію в рядку, а відповідно алель – це значення, якого він набув (0 чи 1). Простим, але наочним прикладом є задача максимізації певної функції з двома змінними (1):

$$f(x_1, x_2) = x_1 * x_2, \quad \text{де} \quad 0 \leq x_1 \leq 1 \quad \text{і} \quad 0 \leq x_2 \leq 1 \quad (1)$$

Як правило, метод кодування справжніх змінних x_1 і x_2 є їх перетворенням на бінарні цілочислові ряди необхідної довжини для забезпечення потрібної точності пошуку в ІС. Допустимо, що десяти розрядів вистачить для кодування x_1 , і x_2 . Встановимо зв'язки між їхнім генотипом і фенотипом закодованих осіб. При цьому поділимо відповідне бінарне подання цілого числа. Отже, [0000000000] відповідає значенню 0/1023 чи 0, а [1111111111] – до 1023/1023 чи 1. Тобто ця структура оптимізації ІС – це 20-бітна рядкова структура, яка представляється у вигляді конкатенації кодів x_1 і x_2 . x_1 знаходиться у 10 крайніх лівих розрядах цієї структури, а x_2 відповідно у крайніх правих розрядах генотипу.

Кодування рішень у системах опрацювання інформації

Авторами було вибрано параметри, їх кількість та кількість розрядів. Потрібно визначитись зі способом запису даних. Існують декілька способів такого кодування. Один з них – класичне бінарне (В) кодування, за якого, наприклад, 10112 = 1110, або ж кодування Грея (G), коли 1011G = 11102 = 1410. Хоча кодування методом Грея потребує специфічного декодування даних, але при цьому дають змогу запобігти деяким проблемам, які виникають під час класичного кодування. Наприклад, очевидною перевагою буде кодування кількох чисел, які ідуть підряд, в такому випадку вони відрізнятимуться лише 1 розрядом. Кодування в код Грея відбувається так:

– з двійкового коду в код Грея: $G[i] = \text{XOR}(B[i+1], B[i])$;

– з коду Грея в двійковий код: $B[i] = \text{XOR}(B[i+1], G[i])$,

де $G[i]$ – і-й розряд у коді Грея, а $B[i]$ – і-й розряд двійкового коду. Отже, числа від 0 до 7 в бінарному коді матимуть вигляд: {000, 001, 010, 011, 100, 101, 110, 111}, а в коді Грея – с{000, 001, 011, 010, 110, 111, 101, 100}.

Формування початкової популяції в інформаційних системах

Класичний алгоритм починає працювати зі створення стартової популяції I_0 – вихідного набору допустимих розв'язків. Вони вибираються випадково або за допомогою якихось простих алгоритмів. Вибір стартових особин для популяції не є важливим. Проте хороша початкова популяція може значно пришвидшити пошук оптимального рішення. Якщо немає припущень, де знаходиться оптимальний розв'язок, то краще формувати початкову популяцію для ІС з особин всього проміжку.

Відбір (селекція)

На будь-якому етапі еволюції ІС використовують ймовірнісний оператор селекції (відбору). Тобто вибирають два рішення-батьки для їх подальшого схрещування. Найпоширенішими серед таких операторів є пропорційна та турнірна селекції. В окремих випадках використовують відбір усіканням. У методі пропорційної селекції (Proportional selection) на k -му кроці можливість вибрати рішення, допустиме для ІС, описується формулою (2):

$$P\{i - \text{вибране}\} = \frac{f(i)}{\sum_{j \in I_k} f(j)}. \quad (2)$$

Ця формула справджується, якщо $f(i) > 0$ для всіх i , які належать I_k (тут I_k – популяція на k -му кроці). Одним з найпростіших способів відбору в ІС є рулетка, за яким вибирають особину за принципом рулетки. Її колесо містить по секторах усіх членів популяції. Розмір такого i -го сектора відповідає величині $P(i)$. З використанням такого відбору члени популяції, які будуть краще пристосовані вибираються частіше, ніж особини з гіршою пристосованістю.

За методом турнірного відбору (Tournament selection) у популяції, яка містить m особин, випадково вибирається t рядків. Рядок, який виявиться найкращим серед них (внаслідок проведення турніру, змагання), записується в тимчасовий масив. Такі дії повторюються m разів. Особини з тимчасового масиву схрещуються (також випадково). У турнірі переважно бере участь парна кількість особин. В такому разі такий турнір називають двійковим/парним.

У разі відбору усіканням (Truncation selection) популяція сортується за зменшенням пристосованості. Кількість можливих особин для схрещування визначається порогом $T \in [0; 1]$. Цей поріг вказує, скільки особин, починаючи з першої (найбільш пристосованої), допускається до відбору. Між такими допущеними до схрещування особинами випадково, $m/2$ разів вибирають батьківські особини, з нащадків яких, своєю чергою, і буде сформовано нову популяцію.

Схрещування в системах опрацювання інформації

Після вибору двох батьківських особин, які піддаються ймовірнісному схрещувальному оператору (crossover), створюються 1 або 2 особини нащадків.

Оскільки з кожної пари батьків утворюється пара нащадків, то для підтримання популяції потрібно схрестити $m/2$ пари. До кожної з таких пар із ймовірністю P_c буде використаний кросовер. Тоді з ймовірністю $1 - P_c$ схрещення не відбувається, і особини у своєму незмінному вигляді переходять на наступний (мутаційний) рівень.

Автори для прикладу наводять елементарний односточковий кросовер, який працює так: на початку випадково вибирають одну можливу точку розриву (ділянку між двома сусідніми бітами), батьківські особини розділяють у цій точці. В процесі кросовера місця розриву двох батьківських особин склеюють між собою для отримання двох генотипів нащадків.

Батько 1 – 1 0 1 1 0 1 1 0 1 1 0 1

Нащадок 1 – 1 0 1 1 0 1 1 0 1 1 1 1

Батько 2 – 0 1 1 0 0 1 1 0 1 1 1 1

Нащадок 2 – 0 1 1 0 0 1 1 0 1 1 0 1

Сьогодні існують й інші варіанти операторів кросоверу: двоточковий та рівномірний кросовер. У двоточковому кросовері вибирається дві точки для розриву, і особини батьків здійснюють обмін сегментами, які розміщені між цими точками. У разі рівномірного оператора схрещування одному з нащадків дістається кожен випадковий біт одного з батьків, а іншому нащадку – випадковий біт іншого батька.

Процес мутації в системах

Після закінчення стадії кросовера структури нащадків піддаються випадковим модифікаціям, або як їх ще називають – мутаціям. У найпростішому випадку кожен біт хромосоми, яка мутує, з імовірністю P_m стає протилежним (таке явище називається одноточковою мутацією). Автори наводять приклад роботи одноточкової мутації:

1 0 0 1 0 1 1 0 0 1 1 1
1 0 0 1 0 1 0 0 0 1 1 1

Складнішим видом мутації є інверсія та транслокація. Інверсія – спосіб мутації, при якому усередині навмання вибираються ділянки хромосоми, де гени поміняються місцями у зворотному порядку:

1 0 0 1 1 1 1 0 0 1 1 1
1 0 0 1 0 0 1 1 1 1 1 1

Транслокація – це мутація, за якої будь-яка ділянка хромосоми переноситься в інший сегмент цієї самої хромосоми:

1 0 0 1 1 1 1 0 0 1 1 1
1 1 1 0 0 0 1 1 0 1 1 1

У певних системах дослідники [16–20] виокремлюють можливість використання механізму інцесту як способу самоадаптації операторів мутації. Його суть у тому, що імовірність мутації будь-якого з генів P_m використовується для усіх його нащадків, спираючись на генетичну близькість його батьківських хромосом. Наприклад, це може бути відношення кількості батьківських генів, які збігаються зі загальною кількістю генів у хромосомі. Такий механізм призводить до цікавого ефекту – за доволі високої різноманітності генофонду цієї популяції наслідки перебігу мутації швидше усього будуть мінімізовані. Це дасть операторові змогу схрещування виконувати своє завдання без втручання зі сторони. У випадку зниження різноманіття хромосом, яке виникає, коли алгоритм призупиняє роботу в оптимумі локальної області, мутаційні наслідки стають відчутнішими. У разі повного сходження популяції алгоритм робиться стохастичним, що ймовірно, своєю чергою, дає змогу популяції вийти з локального оптимуму зі значно більшою кількістю. Для підвищення середньої пристосованості популяції допускаються здійснення мутаційного направлення. Тобто після будь-якої зміни у хромосомі виконується перевірка: чи підвищилась її пристосованість внаслідок здійснення цієї мутації. Якщо ж ні, то хромосома повертається до початкового стану.

Створення нового покоління в системі

Після схрещування і мутації особин потрібно вирішити проблему вибору серед нових особин їх входження у нове покоління та що робити з особинами батьків. Автори розглядають два найпоширеніші способи:

1. Новостворені особини (нащадки) стають на місця їхніх батьків. Потім проходить наступний етап, на якому особини нащадків оцінюються – здійснюється їх відбір, тобто батьки віддають свої місця “дітям”.

2. Наступна створена популяція містить батьківські особини та особини нащадків. У другому випадку потрібно визначити, хто саме з батьків і дітей будуть присутні у новому поколінні. У найпростішому випадку після кожного етапу схрещування у нове покоління включаються дві кращі особини з чотирьох батьків та їхніх дітей. Ефективним є витіснення “схожих” хромосом з популяції та зберігання у ній лише тих, які відрізняються.

Принцип “елітизму”

Цей принцип полягає у включенні в нове покоління кращих батьківських особин. Застосування принципу “елітизму” дозволяє системі не загубити краще рішення на проміжному етапі, але водночас це може бути причиною “застрягання” алгоритму в локальному екстремумі. “Елітизм” не є перешкодою для пошуку вирішення проблеми. Принцип надає алгоритму можливість аналізувати різні хромосоми з простору пошуку.

Зупинка алгоритму

Робота ГА є процесом ітерації, що продовжується доти, поки циклом не виконається перевірка заданої кількості поколінь до умови зупинки алгоритму. В задачах оптимізації

стандартними критеріями для зупинки алгоритму є або довготривала відсутність прогресу, тобто покращення середнього (або кращого) результату пристосованості популяції, чи, наприклад, недостатньо велика різниця між поточним і отриманим внаслідок виконання цих ітерацій результатів пристосованості.

Моделі генетичних алгоритмів системи

1. Canonical GA (J. Holland) – модель з класичним механізмом, яку запропонував Джон Холланд у своїй праці “Адаптація в природних і штучних середовищах” (1975). Ця модель ГА характеризується певними параметрами:

- визначений популяційний розмір та генна розрядність;
- відбір відповідно до пропорцій;
- односточковий кросовер і мутація;
- відсутність “елітизму” при формуванні поколінь.

Алгоритм роботи ГА за такої моделлю матиме такий вигляд:

1. Задання стартової популяції. Візьмемо номер епохи $t = 0$. Задамо випадково m генотипів особин і сформуємо із них популяцію. Обчислимо пристосованість кожної з особин цієї популяції $F(0) = (f_1(0), \dots, f_m(0))$, а потім у середньому пристосованість усієї популяції.

2. Вибір батьківських особин для схрещування. Збільшимо поточний номер епохи на одиницю: $t = t + 1$. Випадково визначимо номер першого батька $l \in \{1 \dots m\}$, задавши імовірність випадання будь-якого числа h пропорційною величині $(t) f(t) f_{cp} h$. Ще одним таким самим випробуванням визначимо номер іншої батьківської хромосоми k .

3. Створення генотипу нащадків. Із заданою імовірністю p_c здійснимо над генотипами вибраних батьківських хромосом односточковий кросовер. Після цього до усіх отриманих даним способом нащадків з вірогідністю p_m застосуємо оператор мутації.

4. Оновлення популяції. Помістимо нащадків у популяцію, заздалегідь викинувши з неї батьківські хромосоми. Обчислимо пристосованість нащадків та, відповідно, змінимо середнє значення пристосованості цієї популяції $(t) f_{cp}$.

Якщо популяції ще не сформовано, повторити усе, починаючи з кроку 2.

2. Genitor (D. Whitley) використовує особливу стратегію для відбору хромосом. Спочатку, як і в інших моделях, популяція створюється та оцінюється. Після цього випадково вибирають двох особин, їх схрещують, причому внаслідок схрещування формується лише одна особина-нащадок, яка оцінюється і стає на місце особини з найнижчим рівнем пристосовування. Потім так само випадково ще раз вибирають 2 особини, і їхній нащадок знову замінює собою особину з найнижчим рівнем пристосованості. За такою моделлю на будь-якому кроці оновлюється лише одна особина з усієї популяції. Автори виділяють такі особливості цієї моделі: визначений розмір поточної популяції; визначена розрядність популяційних генів; особини для схрещування вибирають випадково; немає обмеження на тип мутації та кросовера; результатом кожного схрещування є лише одна особина, яка займає місце найменш пристосованої.

3. Hybrid algorithm (L. "Dave" Davis) дає змогу поєднати плюси ГА з перевагами стандартних методів. ГА дозволяє знайти найкраще рішення для системи, але пошук оптимального розв'язку часто виявляється нелегким завданням через стохастичність принципів роботи цього алгоритму. Внаслідок цього з'явилась ідея використовувати ГА на початкових кроках для ефективного зменшення простору пошуку навколо екстремуму. Потім, вибравши найкращу особину, слід використати “класичний” метод оптимізації. Автори виділяють такі характеристики гібридного алгоритму:

- визначений розмір популяції;
- визначена генна розрядність;
- комбінація стратегій відбору та створення наступного покоління;
- відсутність обмежень типів кросовера та мутації;
- спочатку використовується ГА, після чого починає працювати один з класичних методів оптимізації.

Еволюційне моделювання є науковим напрямом, який сьогодні набирає швидких обертів у галузі математичного моделювання. В ньому об'єднано такі методи моделювання еволюційних процесів, як генетичні алгоритми, еволюційні стратегії та програмування. Його використання

полегшує визначення вихідного рішення з одного чи декількох альтернативних рішень за допомогою стохастичних, детермінованих чи комбінованих алгоритмів. Порівнюючи звичайні оптимізаційні методи та еволюційні алгоритми, можна виділити їхні особливості – це паралельний пошук, випадкові мутації та рекомбінації готових рішень. Застосування генетичних алгоритмів допомагає розв'язувати задачі комбінаторної оптимізації. Вони є адаптивними методами пошуку, які використовують для розв'язання задач функціональної оптимізації.

Висновки та перспективи подальших наукових розвідок

Отже, методи еволюційного моделювання можна розглядати не тільки як ефективні засоби вирішення завдань оптимізації, самоорганізації і моделювання адаптивної поведінки в різних прикладних областях, але і як базу для вправ у вдосконаленні технік програмування окремих алгоритмів та програмних систем. Особливе місце тут займає генетичний алгоритм, який містить всі існуючі лінгвістичні конструкції сучасних мов програмування (зокрема можливості паралельного програмування). ГА є відправною точкою для створення його модифікацій – нових генетичних алгоритмів для розв'язання спеціальних прикладних задач. Таким чином, технологія еволюційного моделювання дозволяє за допомогою використання природних методів еволюції створити розгорнуті інформаційні системи та алгоритми, причому маючи обмежений набір вхідних даних та параметрів. При цьому із застосуванням технології об'єктно-орієнтованого програмування можна реалізувати перелічені вище задачі максимально гнучко і зручно для програміста. Такі технології є ефективними за подальшого їх впровадження та супроводження в ІС.

1. Rosenberg R. S. *Simulation of genetic populations with biochemical properties* // *Mathematical Biosciences*. – 1970. – Volume 8, Issues 1–2, P. 1–37.
2. Schaffer J. D. *Multiple objective optimization with vector evaluated genetic algorithm* / Schaffer J. D. // *Proc. of the First Int. Conf. on Genetic Algorithms*. – NJ: L. Erlbaum, 1985. – P. 93–100.
3. Андрейчиков А. В. *Интеллектуальные информационные системы* / А. В. Андрейчиков, О. Н. Андрейчикова. – М.: Финансы и статистика, 2004. – 424 с.
4. Білоус Р. В. *Особенности прикладного застосування генетичного алгоритму пошуку оптимальних шляхів на графі* / Р. В. Білоус, С. Д. Погорілий // *Реєстрація, зберігання і оброб. даних*. – 2010. – Т. 12, № 2. – С. 81–87.
5. Колесніков К. В. *Генетичні алгоритми для задач багатокритеріальної оптимізації в мережах адаптивної маршрутизації даних* / К. В. Колесніков, А. Р. Каранетян, Т. А. Царенко // *Вісник НТУ “ХП”*. Серія: Нові рішення в сучасних технологіях. – 2013. – № 56 (1029). – С. 44–50.
6. *Генетичні алгоритми, штучні нейронні мережі і проблеми віртуальної реальності* / Г. К. Вороновський та ін. – Харків: Основа, 1997. – 112 с.
7. Гладков Л. А. *Генетические алгоритмы* / Л. А. Гладков, В. В. Курейчик, В. М. Курейчик. – М.: Физматлит, 2010. – 317 с.
8. Задорожна Н. Т. *Підхід до проектування систем моніторингу як складової інформаційних технологій підтримки діяльності органів державного управління* / Н. Т. Задорожна, О. О. Каплун, Т. В. Кузнєцова // *Проблеми програмування*. – 2002. – № 1–2. – С. 368–377.
9. Ісаєв С. А. *Популярно про генетичні алгоритми* / С. А. Ісаєв. – Режим доступу: <http://algotlist.manual.ru/ai/ga/ga1.php>.
10. Каширина И. Л. *Введение в эволюционное моделирование: учеб. пособие* / И. Л. Каширина. – Воронеж, 2007. – 40 с.
11. Колесніков К. В. *Методи адаптивної і нейромережевої маршрутизації пакетів даних в мультиагентних комунікаційних системах* / К. В. Колесніков, Е. В. Кулинич // *Вісник ЧДТУ*. – 2008. – № 2. – С. 5–8.
12. Кормен Т. *Алгоритмы: построение и анализ* / Т. Кормен, Ч. Лейзерсон, Р. Ривест. – М.: Центр непрерыв. математического образования, 2000. – 960 с.
13. Погорілий С. Д. *Генетичний алгоритм розв'язання задачі маршрутизації в мережах* / С. Д. Погорілий, Р. В. Білоус // *Проблеми програмування*. – 2010. – № 2–3. – С. 171–178.
14. Рутковская Д. *Нейронные сети, генетические алгоритмы и нечеткие системы* / Д. Рутковская, М. Пилинский, Л. Рутковский. – М.: Горячая линия Телеком, 2008. – 452 с.
15. Седжвик Р. *Фундаментальные алгоритмы на C++*. *Алгоритмы на графах; пер. с англ.* / Р. Седжвик. – СПб.: ООО “ДиаСофтЮП”, 2002. – 496 с.
16. Сетлак Г. *Решение задач многокритериальной оптимизации с использованием генетических алгоритмов* / Сетлак Г. // *System Research and Information Technologies*. – Kiev: IASA National Academy of Sciences and Ministry of Education and Science Ukraine, 2002. – № 3. – P. 32–42.
17. Стариков А. *Лабораторія BaseGroup. Генетичні алгоритми – математичний апарат* / А. Стариков. – Режим доступу: <http://www.basegroup.ru/genetic/>
18. Фогель, Л. *Штучний інтелект та еволюційне моделювання* / Л. Фогель, А. Оуенс, М. Уолш. – М.: Мир, 1969. – 230 с.
19. Черноруцкий И. Г. *Методы принятия*

решений / И. Г. Черноруцкий. – СПб.: БХВ-Петербург, 2005. – 416 с. 20. Ямина Б. Генетические алгоритмы / Б. Ямина. – СПб.: Изд-во СПб. ун-та, 2005. – Режим доступа: <http://rain.ifmo.ru/cat/view.php/theory/unsorted/genetic-2005>.

УДК 007 : 304 : 001

Н. Е. Кунанец¹, Р. М. Небесний², О. В. Мацюк²

¹ Національний університет “Львівська політехніка”,
кафедра інформаційних систем та мереж,

² Тернопільський національний технічний університет ім. І. Пулюя,
кафедра комп’ютерних наук,
факультет комп’ютерно-інформаційних систем і програмної інженерії

ОСОБЛИВОСТІ ФОРМУВАННЯ ЦІЛЕЙ СОЦІАЛЬНИХ ТА СОЦІОКОМУНІКАЦІЙНИХ СКЛАДОВИХ У ПРОЕКТАХ “РОЗУМНИХ МІСТ”

© Кунанець Н. Е., Небесний Р. М., Мацюк О. В., 2016

Наведено результати пошуку та аналізу публікацій, в яких досліджено вплив “людського фактора” на формування цілей у соціокомунікаційній складовій портфеля проектів “Розумне місто”, оскільки ключовим елементом розвитку систем “розумне місто” є соціум з його потребами та перспективами. Саме люди, а не технології є творцями розвитку сучасних міст, активними учасниками проектів з їх трансформацій та основними споживачами надаваних міських послуг. Автори проаналізували: особливості формування центрів інноваційних адміністративних послуг, що ґрунтуються на нових моделях взаємодії громадян та влади, трансформації міських громад за громадяноцентристським підходом, який сприяє успішному вирішенню проблем міста внаслідок інноваційних перетворень та практичної реалізації розумних рішень на основі інформаційних технологій.

Ключові слова: розумні міста, люди, населення, інновації, керування, потреби, формування цілей.

The article presents results of the search and analysis of publications, which investigated the influence of the 'human factor' on the formation of goals in the socio-communicative section of the portfolio of projects 'Smart City', as a key element in the development of systems "smart city" is the society with its needs and perspectives. It is people, and not technology, who are the creators of modern cities, active participants of projects with their transformations and the major consumers of urban services. The authors analyzed the peculiarities of formation of centers of innovation of administrative services based on new models of interaction between citizens and the authorities, and the transformation of urban communities, with a citizen-centric approach, which contributes to the successful solution of problems of the city through innovative transformation and practical implementation of intelligent solutions based on information technologies.

Key words: social, humans smart city, societal needs, human resources, urban analytics, urban GIS, urban analysis and modeling, social innovation, populations smart city, creative city,

Вступ. Загальна постановка проблеми

Стрімкий розвиток інформаційного суспільства, поступальний перехід до суспільства, яке ґрунтується на знаннях, формує потребу створення ефективного соціокомунікаційного середовища кожної територіальної громади. Особливої актуальності ця проблема набуває в контексті міського