

ОСОБЛИВОСТІ РОЗРОБЛЕННЯ ВЕБ-СИСТЕМ ДЛЯ ЕВОЛЮЦІЙНИХ ОБЧИСЛЕНЬ

© Кривий Р., Ткаченко С., Теслюк В., Зелінський А., 2011

Описано особливості генетичних алгоритмів, а також зроблено огляд програмних реалізацій генетичних алгоритмів. На основі цих реалізацій розглянуто особливості розроблення систем для еволюційних обчислень, а також описано особливості реалізації веб-системи.

Ключові слова: еволюційні обчислення, веб-система, генетичні алгоритми, клієнт-сервер.

The article describes the features of genetic algorithms, as well as a review of program implementation of genetic algorithms. Based on these realizations discusses the features of systems development for evolutionary computation, and describes the features of web-system.

Keywords: evolutionary computing, web system, genetic algorithms, client-server.

Вступ

Залишається актуальним пошук оптимальних розв'язків задач різної природи. Важливість прийняття оптимальних рішень зростає, підвищується відповідальність за прийняті рішення, а їхні наслідки стають значимішими. Отже, з'являється необхідність у методах, інструментах і насамперед, у комп'ютерних програмах, які дають можливість людині знайти найприйнятніші рішення серед різних альтернатив. Надійною базою для таких програм є генетичні алгоритми еволюційного типу.

Особливості генетичних алгоритмів

Все частіше дослідники починають використовувати еволюційні обчислення для розв'язання різноманітних задач, які неможливо вирішити за прийнятний час з використанням традиційних методів. Крім того, є багато проблем, які неможливо вирішити традиційними методами, що обумовлює важливість розроблення і дослідження алгоритмів еволюційного типу.

Генетичний алгоритм – це еволюційний алгоритм пошуку, що використовується для розв'язання задач оптимізації і моделювання шляхом послідовного підбору, комбінування і варіації необхідних параметрів з використанням механізмів, що нагадують біологічну еволюцію.

Генетичні алгоритми відрізняються від інших оптимізаційних і пошукових процедур наступним [1, 2]:

- працюють переважно не з параметрами задачі, а із закодованою множиною параметрів;
- здійснюється пошук не шляхом покращення одного рішення, а шляхом використання одразу декількох альтернатив на заданій множині рішень;
- використання значень цільової функції, а не її різних приростів, для оцінювання якості прийняття рішень.

Враховуючи ці особливості, розроблення системи генетичного пошуку передбачає такі основні етапи: розроблення структури; вибір або розроблення принципу кодування і декодування хромосом; розроблення основних генетичних операторів; розроблення загальної структури генетичного пошуку [3–5].

Існуючі програмні реалізації генетичних алгоритмів

Програмні продукти у цій галузі можна розділити на декілька великих категорій [1].

Перша категорія програм – пакети, що реалізують класичний генетичний алгоритм з можливим налаштувань параметрів управління основними операторами генетичних алгоритмів. Модель хромосоми в таких пакетах має, як правило, стандартну бінарну структуру, а функцію відбору задано одним математичним виразом. До таких програм належать: пакет Evolver 4.0 компанії Palisade Corp [6], пакет GeneHunter компанії Ward System Group [7] тощо. Істотним недоліком цих методів є прив'язка до бінарної або числової моделі хромосоми (хромосома не може мати складної структури).

До другої категорії програм належать спеціалізовані програми, призначені для вирішення конкретних завдань. Ці генетичні алгоритми розроблені і оптимізовані для вирішення вузької, чітко визначеної проблеми.

Третя категорія розробок з генетичних алгоритмів містить наукові дослідження властивостей і характеристик різних генетичних алгоритмів, їх збіжностей і виродженості.

Ефективність генетичного алгоритму під час розв'язання конкретної задачі залежить від багатьох чинників, зокрема таких, як генетичні оператори і вибір відповідних значень параметрів, а також способу представлення рішення на хромосомі. Оптимізація цих чинників приводить до підвищення швидкості і стійкості пошуку, що істотно впливає на застосування генетичних алгоритмів.

На розроблення, програмну реалізацію і результат роботи генетичних алгоритмів впливають такі фактори:

- Програмна реалізація кожного генетичного алгоритму відбувається практично з нуля, тому всі напрацювання з попередніх проектів переважно втрачаються.
- Розробники часто використовують лише найпоширеніші схеми генетичних алгоритмів і генетичних операторів, після чого розробляють спрощені варіанти застосування генетичних алгоритмів.

Найчастіше результати розроблення генетичних алгоритмів відкидаються (проектні рішення, програмну реалізацію тощо). Така ситуація зумовлена відсутністю достатньо гнучких і зручних інструментів проектування, загальноприйнятих стандартів реалізації генетичних алгоритмів і форматів даних, різним рівнем кваліфікації програмістів, що, своєю чергою, унеможливує повторне використання існуючого програмного коду. Крім того, імовірнісний характер генетичного алгоритму обумовлює малу повторюваність результатів. Сукупність названих обставин ускладнює дослідження ефективності і тимчасової складності генетичних алгоритмів, а також робить зіставлення результатів різних алгоритмів надзвичайно трудомістким і необ'єктивним [8,9].

Отже, щоби розробити систему для побудови генетичних алгоритмів, потрібно врахувати всі ці проблеми. Запропонована система повинна мати максимально широку універсальність, підтримувати великий діапазон концепцій побудови генетичних алгоритмів, забезпечувати можливість розширення функціональності. З іншого боку, універсальність і широта можливостей не повинна накладати істотних вимог до обчислювальних ресурсів.

Перспективи розвитку програмного забезпечення

Аналіз сучасного стану програмного забезпечення (ПЗ) показав, що програми можна поділити на декілька категорій з такими характерними рисами:

- Клієнтські програми встановлюються на кожному машині користувача окремо;
- Клієнтські програми встановлюються на кожному машині користувача окремо, а опрацьовуються дані на окремому сервері (використовують клієнт-серверну технологію).
- Як клієнтську програму використовують web-браузер, сервером додатків є web-сервер.

Найперспективнішими є системи, які створено саме на основі технології використання веб-браузера. Це викликано тим, що всесвітня павутина стала глобальною системою зв'язку для доставки інформації та послуг, а серед програмного забезпечення веб-додатки стали однією з галузей, в якій спостерігається найбільше зростання. Такі системи мають багато переваг [3]:

- зручність у разі оновлення ПЗ: оновити ПЗ можна лише на сервері, що потребує менше часу та зусиль, а також полегшує супровід системи;
- легкість масштабування: для того, щоб запустити програму, не потрібно встановлювати будь-яке додаткове ПЗ. Усе, що потрібно, – веб-браузер, який є у будь-якій операційній системі (ОС), та доступ до сервера за допомогою локальної мережі чи інтернету;
- кросплатформеність — система не залежить від типу операційної системи, яку встановлено на машині користувача.

Для реалізації таких систем найкраще підходить Adobe AIR-технологія – незалежне від платформи операційне середовище. Програма, написана з використанням AIR, може виконуватися не лише у браузері, а й як звичайна десктопна програма. AIR дає можливість перетворити існуючі веб-сервіси, написані із використанням Flash, Actionscript, HTML або JavaScript, на традиційні програми для ПК. Зазвичай веб-сервіси зберігають дані користувача на власних серверах. З іншого боку, можливість зберігати свою інформацію на власному ПК часто дуже важлива для користувача. Також AIR-додатки мають змогу працювати без під'єднання до мережі Інтернет. Додаток, написаний з використанням AIR, може бути запущений на декількох платформах, для яких Adobe або її партнери поставляють середовище виконання, а саме: Microsoft Windows NT (XP, Vista, і 7), Mac OS X (PowerPC і Intel), Linux, Android.

Переваги цієї технології:

- Використовуючи AIR, можна легко перенести готовий HTML або Adobe Flex додаток на комп'ютер користувача;
- Додатки мають доступ до файлової системи, буферу обміну і технології drag and drop.

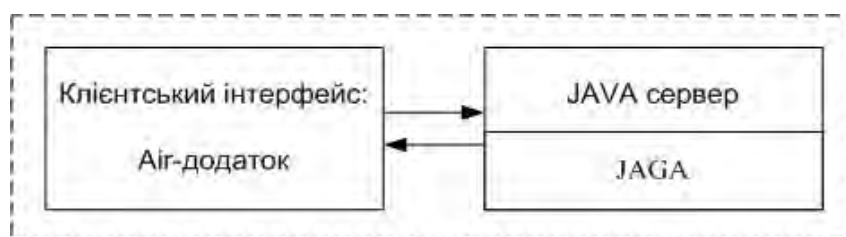
Недоліки:

- Обмежений доступ до веб-сервісів;
- Додатки, написані на AIR, є однопоточними.

Розроблення веб-системи GAWeb

Проаналізувавши найбільш популярні бібліотеки в області еволюційних обчислень, які описані в [10], особливості технології AIR[11], було реалізовано веб-аплікацію GAWeb. В її основу було покладено бібліотеку JAGA (Java API for Genetic Algorithms) – це набір генетичних алгоритмів і операторів, які виконуються на веб-сервері. Серверна частина також написана мовою Java.

Обов'язки клієнтської і серверної частини чітко поділено. Клієнтська частина відповідає за збирання інформації про параметри від користувача і за відображення результатів виконання в певному форматі, а серверна частина – за обчислення (рисунок).



Клієнт-серверна модель системи GAWeb

Сама програма починає працювати в стандартному браузері, завантажує веб-сторінку, на яку, своєю чергою, завантажується вбудований аіг-додаток. У процесі роботи додатка код передається через Інтернет з веб-сервера на браузер клієнта, де він інтерпретується. Користувач вводить всю необхідну інформацію, яка потрібна для роботи генетичного алгоритму, через стандартні компоненти графічного інтерфейсу, такі як розкриті списки, текстові поля і т.д. Після запуску генетичного алгоритму, на основі введеної інформації формується XML-файл, який відправляється на сервер. Сервер збирає інформацію про параметри від клієнта, і, своєю чергою, викликає метод, який відповідає за виконання генетичного алгоритму.

Розроблена веб-система завдяки своїм перевагам дасть змогу спростити використання генетичних алгоритмів для оптимізаційних задач. Вона має такі переваги: зручність у разі оновлення ПЗ, легкість масштабування, кросплатформеність, доступ до системи з будь-якого комп'ютера глобальної мережі. Розвиток цієї системи полягає у подальшому розширенні типів задач, які можна розв'язувати за допомогою генетичних алгоритмів.

Висновок

З кожним роком кількість бібліотек і розроблених систем на основі еволюційних алгоритмів збільшується, а їхні функціональні можливості вдосконалюються. Але під час реалізації програмної системи з використанням розроблених бібліотек слід пам'ятати, що ефективність генетичного алгоритму при розв'язанні конкретної задачі залежить від багатьох чинників і, зокрема, від таких, як генетичні оператори і вибір відповідних значень параметрів, а також способу подання розв'язку на хромосомі. Оптимізація цих чинників приводить до підвищення швидкості і стійкості пошуку, що істотно впливає на застосування генетичних алгоритмів.

1. Kryvyy R. *Analysis of existent systems in researching genetic algorithms* / R. Kryvyy, M. Lobur, S. Tkatchenko // *Proc. of the XV Ukrainian-Polish CADMD. – Krasiczyn (Poland), 2009. – P. 24-25.*
2. Курейчик В. М. *Генетические алгоритмы* / Курейчик В. М. – Таганрог.: ТРТУ, 1998, 239 с.
3. Lobur M. *System's structure design for genetic search* / M. Lobur, S. Tkatchenko, R. Kryvyy, I. Darnobyt // *Proc. of the 5th International Conference of Young Scientists MEMSTECH. – Lviv–Polyana, 2009. – P. 60.*
4. Кривий Р. З. *Розробка підсистеми для дослідження генетичних алгоритмів з використанням шаблонів* / Р. З. Кривий, М. М. Лобур, С. П. Ткаченко // *Вісник Нац. ун-ту “Львівська політехніка” “Комп'ютерні системи проектування. Теорія і практика.” – 2009. – №651. – С. 182–186.*
5. Теслюк В. М. *Розроблення підсистеми для розв'язання оптимізаційних задач еволюційними методами* // В. М. Теслюк, Р. З. Кривий, Тарік (Мох'д Тайсір) Алі АльОмарі, Т. М. Теслюк // *Збірник наук.-техн. праць. Науковий вісник НЛТУ України. – 2009. – Вип. 19.4. – С. 243–249.*
6. <http://www.palisade.com/Evolver/>
7. <http://www.wardsystems.com/GeneHunter.asp>
8. Бакало М. А. *Концепция построения системы поддержки генетических алгоритмов* / М. А. Бакало, В. В. Курейчик // *Перспективные информационные технологии и интеллектуальные системы. – 2007. – № 1. – С. 4–10.*
9. Курейчик В. В. *Возможности организации интегрированной инструментальной среды поддержки процедур генетического поиска и оптимизации решений* // В. В. Курейчик, Е. В. Нужнов // *Известия Таганрогского государственного радиотехнического университета. – 2003. – № 2. – С. 71–79.*
10. Rostyslav Kryvyy. *Analysis of Frameworks for Developing Genetic Algorithms* / Rostyslav Kryvyy, Serhii Tkachenko, Volodymyr Karkuljovskyy. *Proc. of the VIIth International Conference MEMSTECH'2011 – Lviv – Polyana, 2011. – pp. 209–210.*
12. <http://www.adobe.com/ru/products/air.html>