

УДК 004.4'242

РАЗРАБОТКА ИНСТРУМЕНТАЛЬНОГО СРЕДСТВА ДЛЯ ГЕНЕРАЦИИ КОНЕЧНЫХ АВТОМАТОВ С ИСПОЛЬЗОВАНИЕМ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ

Е.А. Мандриков, В.А. Кулев

(Санкт-Петербургский государственный университет информационных технологий, механики и оптики)

В рамках проведения научно-исследовательской работы выполняется разработка инструментального средства для генерации автоматов на основе генетического программирования, которое позволяет повысить уровень автоматизации проектирования автоматных программ. На английском языке это инструментальное средство имеет аббревиатуру – GAAP (Genetic Algorithms for Automata-based Programming).

Ключевые слова: генетические алгоритмы, инструментальное средство, конечный автомат, автоматное программирование

Введение

В настоящее время предложено большое число различных реализаций схем генетических алгоритмов [1–3], а также генетических операторов и способов кодирования особей. Многие существующие генетические алгоритмы являются чисто теоретическими разработками, которые не поддержаны программными продуктами. Другие опробованы на тестовых задачах, но их применение в решении практических задач затруднено из-за отсутствия или недостаточной гибкости соответствующих программных систем [4].

Перечислим недостатки, которыми обладают известные программные средства, реализующие генетические алгоритмы.

1. Жесткая привязка разработанного программного обеспечения к задаче на этапах кодирования и декодирования особей.
2. Программная реализация генетического алгоритма производится практически с нуля.
3. Закрытость разработанных программ как для доработки, так и для интеграции с другими программами.
4. Невозможность сохранения результатов поиска и промежуточных состояний популяции и, следовательно, невозможность анализа этой информации в дальнейшем.

Отметим, что в теореме NFL (No Free Lunch) [5] утверждается, что у всех алгоритмов, которые ищут экстремум функции качества, производительность одинакова, если усреднить результаты по всем возможным функциям качества. Практическое значение этой теоремы состоит в том, что не существует панацеи на все случаи жизни, а несомненный успех какого-либо оптимизационного метода в определенной области знаний не гарантирует такого же успеха в другой области. Это означает, что для каждой специфической области необходимо проводить исследования и отыскивать тот оптимизационный метод, который подходит для нее наилучшим образом. При этом можно выделить ряд особенностей систем, использующих генетические алгоритмы.

1. Для каждой оптимизационной задачи целесообразно строить свою функцию приспособленности [1, разд. 2.3 и 2.4]. Следовательно, система должна обладать возможностью использования различных функций приспособленности.
2. В систему должны входить генетические операторы (создание случайной особи, мутация, рекомбинация, переупорядочение). Напомним, что для различных представлений структур хромосом необходимы различные операторы, учитывающие специфику представления.

3. Необходимым условием эффективности работы систем с использованием генетических алгоритмов является ее автоматическая или автоматизированная настройка на объект исследования.

Исходя из изложенного, сформулируем ряд требований к разрабатываемому инструментальному средству для поддержки генетических алгоритмов, генерирующих конечные автоматы:

1. наличие встроенных типов кодирования хромосом и, как следствие, библиотек генетических операторов;
2. наличие блока подбора и адаптации параметров генетического алгоритма и генетических операторов под решаемую задачу;
3. возможность расширения за счет использования внешних функций генетических операторов, способов кодирования и функций пригодности;
4. возможность визуализации получаемых решений;
5. многокомпонентность системы с возможностью включения и отключения требуемых для решаемой задачи компонентов.

Описание разрабатываемого инструментального средства

Для реализации прототипа инструментального средства был выбран язык Java (<http://java.sun.com/>). Простота интеграции полученной системы с другими системами достигается за счет использования модульности и автоматизированного средства сборки Maven (<http://maven.apache.org/>).

Инструментальное средство состоит из следующих компонентов.

1. Модуль поддержки генетических алгоритмов, который включает в себя:
 1. репозиторий особей для сохранения промежуточных и конечных результатов работы. Репозиторий можно использовать в качестве общего хранилища особей при реализации метода композитного генетического программирования [6]. Использование нескольких репозиториев позволяет реализовать островную модель генетических алгоритмов [1] с различной топологией и способами обмена особями между островами, которую в дальнейшем предполагается включить в модуль;
 2. реализацию классического генетического алгоритма с блоком подбора и адаптации параметров алгоритма и генетических операторов под решаемую задачу;
 3. средства для различных типов визуализаций и отладки процесса выполнения генетического алгоритма. Пример визуализации приведен на рис. 1;
 4. различные операторы и способы представления хромосом (бинарное и вещественное кодирование [7]) для задач численной оптимизации.
2. Модуль поддержки конечных автоматов, который включает в себя:
 1. различные способы представления конечных автоматов. В настоящее время реализованы два представления (строки фиксированной длины и деревья решений [8], и будет внедрено еще одно (сокращенные таблицы [9]);
 2. поддержку XML-представления, которое позволяет хранить и преобразовывать автоматы. Предлагаемое XML-представление является основой для построения различных визуальных представлений автомата с помощью XSL-преобразования. Указанное представление может быть конвертировано в коды на различных языках программирования [10];
 3. реализацию модели «конечный автомат + объект управления = автоматизированный объект» [11], позволяющей конечному автомату воздействовать на объект управления;

4. механизм визуализации процесса работы автоматизированного объекта с возможностью выполнения по шагам, как в прямом направлении, так и в обратном направлении. Пример визуализации приведен на рис. 2.
3. Модуль поддержки генетического программирования для конечных автоматов, который включает в себя набор встроенных в систему генетических операторов для различных представлений конечных автоматов в виде хромосом, реализованных в указанном выше модуле. Также планируется внедрение в модуль поддержки генетических алгоритмов функциональности для распределенных вычислений функции приспособленности, так как эти вычисления очень трудоемки.

После реализации базовых компонентов системы полученный программный комплекс был апробирован на ряде тестовых задач: задача о построении «разливочной линии» [9], задача о «флибах» [1, 12, 13], задача об «умном муравье» [1, 14]. В дальнейшем планируется применение полученного программного средства для решения задач программирования мобильных роботов.

Следует также отметить, что проект GAAP разрабатывается в рамках движения за открытую проектную документацию (<http://is.ifmo.ru/foundation/?i0=foundation>) и является проектом с открытым кодом (сайт проекта GAAP – <http://godin.net.ru:8080/projects/gaap/>), что способствует его дальнейшему развитию.

Заключение

Ранее реализованное инструментальное средство [14] было ориентировано на исследование лишь одной задачи («Умный муравей»), что затрудняло его дальнейшее расширение и использование на других задачах. Другое инструментальное средство [9] было ориентировано на использование лишь одного представления конечных автоматов, что не давало возможности тщательно изучать решаемую задачу.

Предлагаемое средство является более гибким и универсальным.

Литература

1. Технология генетического программирования для генерации автоматов управления системами со сложным поведением. Отчет по I этапу темы.
2. Технология генетического программирования для генерации автоматов управления системами со сложным поведением. Отчет по II этапу темы.
3. Фогель Л., Оуэнс А., Уолш М. Искусственный интеллект и эволюционное моделирование. – М.: Мир, 1969.
4. Технология генетического программирования для генерации автоматов управления системами со сложным поведением. Отчет о патентных исследованиях.
5. Wolpert D. H., Macready W. G. No free lunch theorems for optimization // IEEE Transactions on Evolutionary Computation. – 1997. – April. – Vol. 1, no. 1. – Pp. 67–82.
6. Бедный Ю.Д. Применение генетических алгоритмов для построения клеточных автоматов.– СПб: СПбГУ ИТМО, 2006.
7. Herrera F., Lozano M., Verdegay J. L. Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis // Artificial Intelligence Review. – 1998. – Vol. 12. – № 4. – P. 265–319
8. Данилов В.Р., Шалыто А.А. Метод генетического программирования для генерации автоматов, представленных деревьями решений. // Научное программное обеспечение в образовании и исследованиях. – СПбГУ ПУ. – 2008. – С. 175–181.

9. Поликарпова Н.И., Точилин В.Н., Шалыто А.А. Разработка библиотеки для генерации управляющих автоматов методом генетического программирования. – 2007 – С. 84–87.
10. Непейвода Н.Н., Скопин И.Н. Основания программирования. – Ижевск-Москва: Институт компьютерных исследований, 2003.
11. Поликарпова Н.И., Шалыто А.А. Автоматное программирование. Учебно-методическое пособие. – СПб.: СПбГУ ИТМО, 2007.
12. Лобанов П.Г., Шалыто А.А. Использование генетических алгоритмов для автоматического построения конечных автоматов в задаче о «флибах» // Известия РАН. Теория и системы управления. – 2007. – №5. – С. 127–136.
13. Мандриков Е.А., Кулев В.А., Шалыто А.А. Применение генетических алгоритмов для создания управляющих автоматов в задаче о «флибах» // Информационные технологии – 2008. – №1. – С. 42–45.
14. Бедный Ю.Д., Шалыто А.А. Применение генетических алгоритмов для построения автоматов в задаче «Умный муравей». – СПбГУ ИТМО, 2007.