

4. Image Transforms – Hough Transform [Electronic resource]. URL: <http://homepages.inf.ed.ac.uk/rbf/HIPR2/hough.htm> (date of visit: 28.11.2012).

5. Tretiak E., Barinova O., Kohli P., Lempitsky V. Geometric image parsing in man-made environments // Intern. J. of Computer Vision. 2012. Vol. 97. № 3. P. 305–321.

6. Barinova O., Lempitsky V., Kohli. P. On the Detection of Multiple Object Instances Using Hough Transforms // IEEE Computer Vision and Pattern Recognition, San Francisco. 2010. P. 1–8.

7. Nixon M., Aguado A. S. Feature Extraction and Image Processing for Computer Vision. 3rd ed. // Academic Press, 2012.

8. Фаворская М. Н., Зотин А. Г., Пахирка А. И. Метод улучшения цветных изображений на основе выравнивания спектральных диапазонов и коррекции контрастности // Механика, управление и информатика. 2012. № 8. С. 99–103.

9. Muthukrishnan R., Radha M. Edge Detection Techniques for Image Segmentation // Intern. J. of Computer Science & Information Technology. 2011. Vol. 3. № 6. P. 259–267.

10. Гонсалес Р., Вудс Р. Цифровая обработка изображений. М. : Техносфера, 2006.

11. Лукьяница А. А., Шишкин А. Г. Цифровая обработка видеозображений. М. : Ай-Эс-Эс Пресс, 2009.

12. Sezgin M., Sankur B. Survey over Image Thresholding Techniques and Quantitative Performance Evaluation // J. of Electronic Imaging. 2004. Vol. 13(1). P. 146–165.

13. Otsu N. A Threshold Selection Method from Gray-Level Histograms // IEEE Transactions on Systems, Man, and Cybernetics. 1979. Vol. 9. № 1. P. 62–66.

© Зотин А. Г., Борисов Ю. В., Лисица А. С., 2013

УДК 519.8

САМОКОНФИГУРИРУЕМЫЙ ГЕНЕТИЧЕСКИЙ АЛГОРИТМ РЕШЕНИЯ ЗАДАЧ ПОДДЕРЖКИ МНОГОКРИТЕРИАЛЬНОГО ВЫБОРА*

И. А. Иванов, Е. А. Сопов

Сибирский государственный аэрокосмический университет имени академика М. Ф. Решетнева
Россия, 660014, г. Красноярск, просп. им. газ. «Красноярский рабочий», 31
E-mail: ilyaiv92@gmail.com, evgenysopov@gmail.com

Предложен новый самоконфигурируемый многокритериальный алгоритм оптимизации на базе коэволюционного генетического алгоритма. Представлены результаты исследования эффективности подхода в сравнении с известными алгоритмами.

Ключевые слова: многокритериальная оптимизация, множество Парето, эволюционные алгоритмы.

SELF-CONFIGURING GENETIC ALGORITHM FOR MULTI-OBJECTIVE CHOICE PROBLEM DECISION

I. A. Ivanov, E. A. Sopov

Siberian State Aerospace University named after academician M. F. Reshetnev
31“Krasnoyarskiy Rabochiy” prospect, Krasnoyarsk, 660014, Russia
E-mail: ilyaiv92@gmail.com, evgenysopov@gmail.com

A novel self-configuring multi-objective algorithm of optimization, based on coevolutional genetic algorithm, is proposed. Results of algorithm efficiency investigation are presented.

Keywords: multi-objective optimization, Pareto set, evolutionary algorithms.

Большое число практических задач поддержки принятия решений в различных областях науки и техники сводятся к решению задачи выбора на множестве альтернатив. Такие задачи возникают на всех этапах поддержки принятия решений: при построении

аналитических моделей (выбор структуры и параметров модели), генерировании множества допустимых альтернатив, непосредственно при выборе наилучшей альтернативы в соответствии с некоторыми критериями. В реальных задачах число критериев, по кото-

* Работа выполнена при финансовой поддержке ФЦП «Научные и научно-педагогические кадры России» (2011-1.2.2-215-021, 2011-1.2.1-113-025), ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007–2013 годы» (2011-1.9-519-005-042).

рым оцениваются альтернативы, почти всегда больше одного.

Формально задача выбора сводится к задаче параметрической оптимизации некоторого функционала, где независимые переменные определяют альтернативы, а значение функционала – оценку качества выбора [1]. В случае с единственным критерием существует возможность нахождения наилучшего решения, обеспечивающего экстремум функционала. В случае многих критериев объективного решения не существует, так как необходимо обеспечить компромисс между конфликтующими критериями, а лицо, принимающее решение (ЛПР), должно осуществить субъективный выбор [2]. На практике исследователи часто упрощают многокритериальную задачу выбора до однокритериальной путем определения важности и свертки критериев. Не трудно понять, что компромисса можно достичь разными способами, а потому нет гарантии, что единственное решение по свертке будет приемлемо для ЛПР. Математически множество эффективных решений многокритериальных задач выбора определяется через понятие оптимальности по Парето [3]. Именно из множества Парето должен осуществляться финальный субъективный выбор, так как любое решение вне множества Парето является менее предпочтительным с математической точки зрения. При этом чем точнее и равномернее локализовано (аппроксимировано) множество Парето, тем эффективнее решена формальная задача многокритериальной оптимизации.

Практические задачи оптимизации обладают такими свойствами, как алгоритмическое задание функций, многоэкстремальность, сложная конфигурация допустимой области, наличие нескольких типов переменных, множества постоянства и т. д. Подобные задачи почти не решаются традиционными методами оптимизации. Проблема усугубляется необходимостью учета требования равномерности покрытия множества Парето. Наиболее перспективным подходом к решению подобных задач являются методы интеллектуальных информационных технологий, к которым, в частности, относятся генетические алгоритмы (ГА) [4–6].

Идея коэволюционного многокритериального генетического алгоритма. В последние годы предложено множество различных генетических алгоритмов, показавших высокую эффективность при решении многих сложных задач оптимизации. Для решения задач многокритериальной оптимизации наибольшее признание в мировом научном сообществе получили следующие алгоритмы: VEGA (Vector Evaluated Genetic Algorithm) [7], FFGA (Fonseca and Fleming's Multi-Objective Genetic Algorithm) (иногда встречается под названием MOGA (Multi-Objective Genetic Algorithm) [8], NPGA (Niche Pareto genetic algorithm) [9], NSGA2 (Non-Dominated Sorting Genetic Algorithm) [9], SPEA (Strength Pareto Evolutionary Algorithm) [10].

Каждый из алгоритмов обладает своими преимуществами и недостатками. В частности, алгоритмы VEGA и FFGA обладают лучшей сходимостью, но не имеют механизмов обеспечения равномерности по-

крытия множества Парето, алгоритмы NPGA, NSGA и SPEA обеспечивают хорошее покрытие, но требуют больше вычислительных затрат, а механизмы поддержания разнообразия решений часто выкидывают решения за пределы области Парето. В целом большинство исследователей отмечают алгоритм SPEA как наиболее универсальный [11].

Процесс настройки и контроля параметров конкретного ГА существенно влияет на эффективность решения задачи. Однако этот процесс слабо формализован и часто зависит от опыта и интуиции исследователя. В последние годы наблюдается развитие эволюционных подходов, способных к самонастройке параметров. Одним из подходов к построению самоорганизующихся (самонастраивающихся) процедур является использование коэволюционного подхода [11; 12].

Большинство работ, связанных с построением коэволюционных схем для многокритериальных ГА, используют схему кооперативной коэволюции [13–15]. Цель подобных алгоритмов – провести декомпозицию исходной задачи по переменным, по критериям, по функциональному назначению популяций и т. д. Оценка пригодности индивидов частного алгоритма (подзадачи) происходит с учетом оценок других индивидов. Подобные схемы коэволюционных ГА демонстрируют весьма высокие результаты, но требуют тщательной настройки как параметров частных алгоритмов, так и схем декомпозиции задачи и кооперации.

При использовании схемы конкурирующей коэволюции различные алгоритмы (популяции) развиваются параллельно и независимо в течение некоторого периода. После происходит оценка их эффективности, и вычислительные ресурсы перераспределяются от менее эффективных (на данном временном интервале) в пользу более эффективных. Также присутствуют элементы кооперативной коэволюции, в частности, миграция лучших индивидов. Таким образом, при использовании конкурирующей коэволюции нет необходимости определять наилучший для данной задачи (класса задач) алгоритм или его настройки – выбор алгоритма (настроек) осуществляется автоматически в ходе работы.

Очевидно, что использование коэволюционной схемы при решении многокритериальных задач позволит решить проблему конфигурирования алгоритма «под задачу». А поскольку различные алгоритмы обладают принципиально различными свойствами, то их совместная работа должна усилить использование индивидуальных сильных сторон и уменьшить влияние слабых. В предложенном подходе в ходе работы поискового алгоритма кроме адаптации численных параметров алгоритма (адаптация алгоритма), но и происходит выбор конкретной конфигурации алгоритма, эффективной на текущем этапе поиска. Такой подход можно назвать самоконфигурируемым многокритериальным ГА.

Авторы назвали такой алгоритм самоконфигурируемым многокритериальным генетическим алгоритмом SelfCOMOGA (Self-configuring Coevolutionary Multi-Objective Genetic Algorithm).

Общая схема коэволюционного алгоритма SelfCOMOGA. Схема имеет следующий вид:

Этап 1. Задание множества алгоритмов, участвующих в коэволюции.

Этап 2. Параллельная работа алгоритмов в течение заданного времени – периода адаптации.

Этап 3. Оценка эффективности индивидуальных алгоритмов в адаптационном периоде.

Этап 4. Перераспределение ресурсов и переход к очередному адаптационному периоду – на этап 2.

Рассмотрим эти этапы подробнее.

На первом этапе определяется множество конфигураций алгоритмов, включенных в коэволюцию. Можно выделить как минимум три стратегии:

- детерминированный выбор алгоритмов, свойства которых априори известны и могут быть использованы для решения задачи;

- включение всех возможных комбинаций алгоритмов и их параметров;

- случайный выбор некоторого числа алгоритмов из множества всех возможных комбинаций алгоритмов и их параметров.

Первая стратегия требует привлечения знаний об алгоритмах и решаемой задаче, что не применимо для сложных задач, и противоречит идее самонастройки ГА. Вторая стратегия – весьма затратна, так как число комбинаций может быть большим (сотни вариантов). Третья стратегия, как показывают численные эксперименты, обеспечивает приемлемую эффективность в среднем и при этом нет необходимости привлекать дополнительные сведения об алгоритмах и их свойствах на задаче (классе задач), а значит, этот вариант лучше всего обеспечивает концепцию самонастройки ГА.

Период адаптации является настраиваемым параметром. Эксперименты показали, что значение этого параметра индивидуально для каждой задачи и зависит от критериев, по которым оцениваются алгоритмы. Например, увеличение параметра дает улучшение по критерию расстояния до множества Парето, но ухудшает показатели покрытия множества Парето и наоборот. Более того, значение параметра определяется ограничением числа вычислений целевых функций (вычислительным ресурсом). Можно отметить, что при достаточном вычислительном ресурсе алгоритм в среднем малочувствителен к данному параметру. Тем не менее, данный вопрос необходимо в дальнейшем исследовать отдельно.

Ключевым моментом любой коэволюционной схемы является оценка эффективности индивидуальных алгоритмов. Поскольку эффективность оценивается по Парето, то нет возможности прямого сравнения алгоритмов и известные подходы не могут быть использованы. В различных работах предлагаются следующие основные критерии: близость найденного множества решений к истинному множеству Парето и равномерность распределения найденных решений. Очевидно, что для практических задач первый критерий теряет смысл, так как множество Парето неизвестно. Поэтому в данной статье для оценки индивидуальных алгоритмов, включенных в SelfCOMOGA, используются следующие критерии, объединенные в две группы.

Первая группа – статические критерии (оценка по результатам только текущего периода адаптации) – включает в себя два критерия:

- процент недоминируемых решений. Для вычисления создается общий пул решений из решений частных алгоритмов и проводится недоминируемая сортировка [8], а затем определяется, какому алгоритму принадлежат недоминируемые решения;

- равномерность распределения (разброс) недоминируемых решений. Для вычисления используется среднее значение дисперсии расстояний между индивидами по координатам (для множества Парето) или по критериям (для фронта Парето).

Вторая группа – динамические критерии (оценка в сравнении с результатами предыдущих периодов адаптации) – включает в себя еще один критерий – улучшение недоминируемых решений. Для вычисления необходимо сравнить недоминируемые решения алгоритма на прошлом и текущем периоде адаптации. Если текущие недоминируемые решения доминируют прошлые недоминируемые, то произошло улучшение решения задачи, даже если процент недоминируемых решений уменьшился.

Последний этап – этап перераспределения ресурсов. Для этого определяются новые размеры популяций. Все алгоритмы отдаются алгоритму-победителю некоторый процент от своих размеров популяции. При этом каждый алгоритм имеет минимальный гарантированный ресурс, который не распределяется.

В известных коэволюционных схемах каждый новый этап адаптации начинается с одинаковых стартовых точек и используется правило «лучший вытесняет худшего». Для многокритериального алгоритма данный подход не пригоден из-за требования обеспечения разнообразия Парето-оптимальных решений. А поскольку лучшие решения обеспечивают сходимость к множеству Парето, то для перераспределения ресурсов используется схема вероятностного отбора «замещение по ранговой селекции»: решения с высоким рангом после недоминируемой сортировки имеют большую вероятность быть переданными лучшим алгоритмам, с уменьшением ранга вероятность отбора линейно падает, но шанс быть отобранным остается у всех решений.

Критерии останова коэволюционного алгоритма аналогичны критериям стандартных ГА: ограничение на вычислительный ресурс, отсутствие улучшений (стагнация) и др.

Тестовые задачи и численные эксперименты. Авторами было проведено исследование эффективности алгоритма SelfCOMOGA на 19 тестовых функциях, которые условно можно объединить в две группы:

- функции 1–6 представляют собой различные тестовые задачи двух переменных. Критерии включают в себя как простые квадратичные функции, так и сложные – (Растригина, Розенброка). Число критериев – от двух до четырех. Данные функции удобны тем, что помимо численных оценок с их помощью можно визуализировать множества и фронт Парето;

- функции 7–19 приняты международным сообществом специалистов по оптимизации на конференции CEC (2009) в качестве сложных задач и рекомендованы

для оценки многокритериальных ГА [16]. Это функции FON, POL, KUR, ZDT1-4, ZDT6, F2, F5, F8, UP4, UP7.

Некоторые результаты тестирования алгоритмов приведены в табл. 1 и 2. Тестирование алгоритма проводилось при следующих условиях. В коэволюцию включены пять различных алгоритмов VEGA, FFGA, NPGA, NSGAII и SPEA, параметры которых выбирались случайно из множества:

Тип скрещивания: одноточечное, двухточечное, равномерное.

Вероятность мутации: высокая, средняя, низкая.

Максимальный размер множества Парето (для SPEA): 30, 50, 70.

Радиус ниши (для FFGA и NPGA): 1, 3, 5.

Размер множества сравнения (для NPGA): 3, 5, 7.

Также в таблицах представлены результаты индивидуальной работы базовых алгоритмов, для которых приведены оценки при лучших параметрах, найденных при индивидуальном тестировании.

На каждой задаче для оценки эффективности проводилось 100 независимых запусков алгоритмов. Частные результаты усреднялись. Статистика обрабатывалась с помощью метода дисперсионного анализа – метода непараметрической оценки Манна–Уитни. Различия результатов, представленных в табл. 1, статистически значимы.

В качестве критериев для сравнения эффективности работы алгоритмов выбраны следующие два набора метрик:

Для задач с аналитически заданным множеством и фронтом Парето (F2, F5, F8, UP4, UP7) использовались общепринятые метрики: Generational Distance (GD), Maximum Spread (MS), Spacing (S).

Критерий GD характеризует различие между найденным (PF) и истинным (PF*) фронтом Парето:

$$GD = \frac{1}{n_{PF}} \left(n_{PF} \sum_{i=1}^{n_{PF}} d_i^2 \right)^{1/2}$$

где $n_{PF} = |PF|$; d_i – евклидово расстояние (в пространстве критериев) между i -м членом PF и ближайшим членом PF*. В идеальном случае $GD = 0$.

Критерий MS – характеризует то, насколько хорошо PF* покрыто PF, большие значения MS показывают, что большая часть PF* покрыта PF:

$$MS = \sqrt{\frac{1}{M} \sum_{i=1}^M \left[\frac{\min \{ \overline{PF}_i, \overline{PF}_i^* \} - \max \{ \underline{PF}_i, \underline{PF}_i^* \}}{\overline{PF}_i^* - \underline{PF}_i^*} \right]^2},$$

где \overline{PF}_i и \underline{PF}_i – максимальное и минимальное значение i -й целевой функции в PF; \overline{PF}_i^* и \underline{PF}_i^* – максимальное и минимальное значения i -й целевой функции в PF*; M – количество целевых функций.

Критерий S – характеризует равномерность распределения недоминируемых решений по фронту Парето:

$$S = \frac{1}{d'} \left(\frac{1}{n_{PF}} \sum_{i=1}^{n_{PF}} (d'_i - \bar{d}')^2 \right)^{1/2},$$

$$\bar{d}' = \frac{1}{n_{PF}} \sum_{i=1}^{n_{PF}} d'_i$$

где $n_{PF} = |PF|$, d_i – Евклидово расстояние (в пространстве критериев) между i -м членом PF и ближайшим членом PF*.

1. Для остальных задач сравнение алгоритмов проводилось относительно друг друга на основе недоминируемой сортировки. Для вычисления значений по критериям решения всех исследуемых алгоритмов объединились в общий пул и проводилась недоминируемая сортировка.

Критерий минимального ранга алгоритма K1 – наименьший ранг, который имеет алгоритм после недоминируемой сортировки в общем пуле.

Критерий доли решения с минимальным рангом K2 – процент популяции, имеющий ранг, определённый критерием K1.

Критерий разброса решений с минимальным рангом K3 – дисперсия расстояний в пространстве критериев для решений, имеющих ранг, определённый критерием K1.

Таблица 1

Оценки эффективности алгоритмов на задачах с аналитически заданным множеством и фронтом Парето (набор метрик 1)

Задача	Критерий	SPEA	VEGA	FFGA	NPGA	NSGAII	Self COMOGA	MEAN
F2	GD	0,849	1,468	3,334	1,699	0,263	0,460	1,522
	MS	0,750	0,768	2,928	0,730	0,850	0,832	1,205
	S	1,289	0,446	0,096	0,400	0,525	1,483	0,551
UP4	GD	0,421	0,441	0,616	0,616	0,426	0,266	0,504
	MS	0,928	0,933	1,127	0,905	0,928	0,973	0,964
	S	0,368	0,612	0,218	0,279	0,340	0,464	0,363
UP7	GD	1,149	1,881	4,683	2,274	0,305	0,629	2,058
	MS	0,719	0,757	1,571	0,713	0,785	0,792	0,909
	S	1,459	0,444	0,092	0,358	0,556	1,735	0,582

Оценки эффективности алгоритмов на задачах, для которых неизвестны аналитические выражения множества и фронта Парето (набор метрик 2)

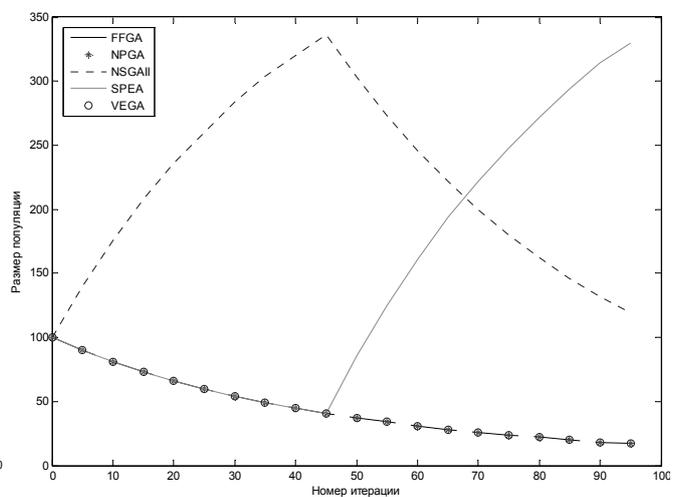
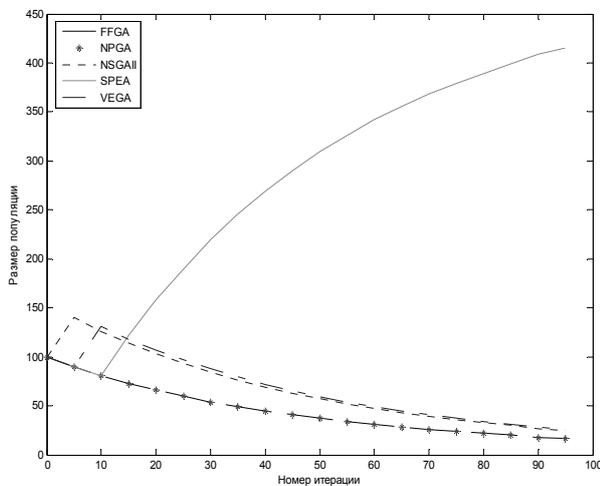
Задача	Критерий	SPEA	VEGA	FFGA	NPGA	NSGAI	Self COMOGA	MEAN
F2 (Schaffer's function)	K1	2	1	26	2	1	1	7
	K2	0,349	0,357	0,610	0,365	0,424	0,330	0,421
	K3	0,000002	0,000002	0,143085	0,000002	0,000001	0,000001	0,029
4 квадратичные функции	K1	1	1	48	2	1	1	11
	K2	0,208	0,158	0,813	0,147	0,277	0,273	0,321
	K3	0,000005	0,000006	0,407079	0,000013	0,000005	0,000005	0,081422
FON	K1	1	1	27	1	1	1	6
	K2	0,196	0,084	0,631	0,087	0,330	0,353	0,266
	K3	0,455	0,504	1,002	0,547	0,394	0,374	0,580
POL	K1	1	1	10	1	1	1	3
	K2	0,228	0,083	0,214	0,062	0,344	0,316	0,186
	K3	0,198	0,202	0,324	0,125	0,133	0,169	0,196
KUR	K1	1	1	21	2	1	1	5
	K2	0,190	0,158	0,620	0,101	0,344	0,358	0,283
	K3	0,691	0,699	0,395	0,676	0,743	0,741	0,641
ZDT1	K1	3	9	26	10	2	1	10
	K2	0,158	0,115	0,318	0,297	0,578	0,993	0,293
	K3	0,239	0,521	0,758	0,455	0,206	0,159	0,436
ZDT6	K1	4	21	48	12	1	1	17
	K2	0,210	0,235	0,627	0,230	0,162	0,948	0,293
	K3	0,458	0,703	1,100	0,655	0,380	0,311	0,659

В табл. столбец MEAN определяет среднее значение критерия по пяти индивидуальным алгоритмам (для сравнения с результатами алгоритма SelfCOMOGA).

Можно отметить, что на множестве задач эффективность коэволюционного алгоритма в среднем может оказаться хуже лучшего из индивидуальных алгоритмов, но всегда выше средней эффективности по индивидуальным алгоритмам. Поскольку в реальных задачах исследователь не знает заранее, какой алгоритм выбрать и какие настройки использовать, то применение самоконфигурируемого коэволюционного ГА кажется весьма перспективным.

Ниже представлены два примера динамики перераспределения ресурсов: для условно простой задачи (квадратичные критерии) – на рисунке слева – и для сложной – справа.

Как видно из графиков, для простых задач метод SPEA может доминировать на протяжении всего времени работы. Однако эффективность коэволюционного алгоритма в среднем оказывается выше, чем у SPEA отдельно. Более того, в коэволюционном алгоритме параметры SPEA выбираются случайно (не оптимально), а значит несмотря на его победу на всех этапах адаптации, общий эффект достигается за счет совместной работы различных алгоритмов.



Динамика распределения ресурсов SelfCOMOGA

Предложенный в данной статье подход позволяет эффективно решать сложные задачи многокритериальной оптимизации, обеспечивая репрезентативную аппроксимацию множества Парето, а использование конкурентной коэволюции позволяет автоматизировать конфигурирование алгоритма под задачу.

Для достаточно простых задач наибольший вклад в коэволюционную динамику вносит алгоритм SPEA, однако эффективность коэволюционного алгоритма в среднем оказывается выше, чем у SPEA самостоятельно, так как при распределении ресурсов все алгоритмы имеют минимальный гарантированный ресурс и дополняют SPEA различными поисковыми стратегиями.

Для более сложных задач преимущества SPEA менее очевидны. В основном этот алгоритм получает больший ресурс на завершающих стадиях, когда требуется поддержка равномерности покрытия множества Парето. На этапе локализации множества Парето ресурс достается алгоритмам, которые с большей скоростью сходятся к парето-оптимальным решениям (в частности, алгоритм VEGA).

В дальнейшем авторы планируют расширить область применения предложенного подхода для автоматизирования проектирования интеллектуальных информационных технологий в задачах интеллектуального анализа данных [17], в которых равномерность покрытия множества Парето обеспечивает разнообразие представлений моделей знаний.

Библиографические ссылки

1. *Taha X. A.* Введение в исследование операций : пер. с англ. 7-е изд. М. : Вильямс, 2005.
2. *Ларичев О. И.* Теория и методы принятия решений. М. : Логос, 2000.
3. *Штойер Р.* Многокритериальная оптимизация. Теория, вычисления и приложения. М. : Радио и связь, 1992.
4. *Goldberg D. E.* Genetic Algorithms in Search, Optimization and Machine Learning. Reading, MA : Addison-Wesley, 1989.
5. Modified Probabilistic Genetic Algorithm for the Solution of Complex Constrained Optimization Problems / F. Yu. Vorozheikin, T. N. Gonchan, I. A. Panfilov et al. // Vestnik SibSAU. 2009. Iss. 5 (26). P. 31–36.
6. *Сопов Е. А.* Вероятностные генетические алгоритмы оптимизации сложных систем // Saarbruecken : Lambert Academic Publishing GmbH & Co. KG, 2012.
7. *Schaffer J. D.* Multiple Objective Optimization with Vector Evaluated Genetic Algorithms // Proc. of an Intern. Conf. on Genetic Algorithms and Their Applications, Pittsburgh, PA, 1985. P. 93–100.
8. *Fonseca C. M., Fleming P. J.* Genetic Algorithms for Multi-Objective Optimization: Formulation, Discussion and Generalization // Proc. of the 5th Intern. Conf. on Genetic Algorithms. San Mateo, California, 1993. P. 416–423.
9. *Horn J., Nafpliotis N., Goldberg D. E.* A Niche Pareto Genetic Algorithm for Multiobjective Optimization // Proc. of the 1st IEEE Conf. on Evolutionary Computation. Piscataway, 1994. Vol. 1. P. 82–87.
10. *Zitzler E., Thiele L.* Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach // IEEE Transactions on Evolutionary Computation. 1999. Vol. 3. №. 4. P. 257–271.
11. *Sergienko R.* Multistep Fuzzy Classifier Forming with Cooperative-Competitive Coevolutionary Algorithm // Advances in Swarm Intelligence : Proc. of the 3rd Intern. Conf. Shenzhen, China, 2012. Pt. I.
12. *Ficici S. G.* Solution Concepts in Coevolutionary Algorithms : A Doctor of Philosophy Diss. Brandeis University, 2004.
13. *Tan K. C.* A Cooperative Coevolutionary Algorithm For Multiobjective Optimization // Proc. of the IEEE Intern. Conf. on Systems, Man & Cybernetics: The Hague, Netherlands, 2004.
14. *Goh C. K.* A Competitive-Cooperation Coevolutionary Paradigm for Multi-Objective Optimization // Proc. of the 22nd IEEE Intern. Symp. on Intelligent Control. Singapore, 2007. P. 255–260.
15. *Zeng F.* Studies on Pareto-Based Multi-Objective Competitive Coevolutionary Dynamics // Proc. of the IEEE Congress on Evolutionary Computation, 2011.
16. Multiobjective optimization Test Instances for the CEC 2009 Special Session and Competition // Proc. of the IEEE Congress on Evolutionary Computation. Norway, 2009.
17. *Сопов Е. А., Сопов С. А.* Интеллектуальные информационные технологии выявления и диагностики проблем в задачах анализа сложных систем // Вестник СибГАУ. 2011. 4 (37). С. 84–89.

© Иванов И. А., Сопов Е. А., 2013