

УДК 519.68

**СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ
МНОГОКРИТЕРИАЛЬНОГО ВЫБОРА НА БАЗЕ КОЭВОЛЮЦИОННОГО
ГЕНЕТИЧЕСКОГО АЛГОРИТМА**

Иванов И.А.

научный руководитель канд. техн. наук Сопов Е.А.

***Сибирский государственный аэрокосмический университет имени академика М.Ф.
Решетнева***

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 которым оцениваются альтернативы, почти всегда больше одного. Как минимум можно выделить следующие два критерия: критерий, связанный с достижением цели выбора (самый явный, очевидный – зависит от решаемой задачи) и экономический критерий, связанный с эффективностью решения задачи (учет ограниченных ресурсов как при решении задачи выбора, так и при решении исходной задачи, когда выбор сделан). Кроме того возникает множество специфичных для частной задачи критериев: надежность, устойчивость, ограничения задачи и т.д.

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 с учетом требования равномерности покрытия множества Парето. Наиболее перспективным подходом к решению подобных задач являются методы интеллектуальных информационных технологий, к которым, в частности, относятся генетические алгоритмы (далее ГА).

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 такие алгоритмы как VEGA, FFGA, NPGA, NSGA2, SPEA.

Каждый из алгоритмов обладает своими преимуществами и недостатками. В частности, VEGA и FFGA обладают лучшей сходимостью, но не имеют механизмов обеспечения равномерности покрытия множества Парето. 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.

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 тенденция развития эволюционных подходов, способных к самонастройке параметров. Одним из

подходов к построению самоорганизующихся (самонастраивающихся) процедур является использование коэволюционного подхода.

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 влияния слабых. В предложенном подходе в ходе работы поискового алгоритма адаптируются не только численные параметры алгоритма (адаптация алгоритма), но и происходит выбор конкретной конфигурации алгоритма, эффективной на текущем этапе поиска. Такой подход можно назвать самоконфигурируемым многокритериальным ГА.

Мы назвали такой алгоритм самоконфигурируемым коэволюционным многокритериальным генетическим алгоритмом или SelfCOMOGA (Self-configuring Co-evolutionary Multi-Objective Genetic Algorithm).

Общая схема коэволюционного алгоритма SelfCOMOGA 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 (перейти на этап 2).

Рассмотрим этапы подробнее. На первом этапе определяется множество конфигурации алгоритмов, включенных в коэволюцию. Можно выделить как минимум 3 стратегии:

- детерминированный выбор алгоритмов, свойства которых известны и можно использовать 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.

Первый вариант требует привлечения знаний об алгоритмах и решаемой задаче, что не применимо для сложных задач, и противоречит идее самонастройки ГА. Второй подход – весьма затратным, т.к. число комбинаций может быть большим (сотни вариантов). Как показывают численные эксперименты, последняя стратегия обеспечивает приемлемую эффективность в среднем, и при этом нет необходимости привлекать дополнительные сведения об алгоритмах и их свойствах на задаче, а значит, этот вариант лучше всего обеспечивает концепцию самонастройки ГА.

Ключевым моментом любой коэволюционной схемы является оценка индивидуальных алгоритмов. Поскольку эффективность оценивается по Парето, 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, т.к. множество Парето неизвестно. Поэтому в данной работе для оценки индивидуальных алгоритмов SMOGA используются следующие критерии, объединенные в 2 группы:

Первая группа - статические критерии (оценка по результатам только текущего периода адаптации) - включает 2 критерия.

K1 – процент недоминируемых решений. Для вычисления создается общий пул решений из решений частных алгоритмов и проводится недоминируемая сортировка [8]. Далее определяется, какому алгоритму принадлежат недоминируемые решения.

K2 – равномерность распределения (разброс) недоминируемых решений. Для вычисления используется среднее значение по координатам (для множества Парето) или по критериям (для фронта Парето) дисперсии расстояний между индивидами.

Вторая группа - динамические критерии (оценка в сравнении с результатами предыдущих периодов адаптации) - включает критерий:

K3 – 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. Для этого все алгоритмы отдают алгоритму-победителю некоторый процент от своих размеров популяции. При этом каждый алгоритм имеет минимальный гарантированный ресурс, который не распределяется.

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 стандартным ГА: ограничение на вычислительный ресурс, отсутствие улучшений (стагнация) и др.

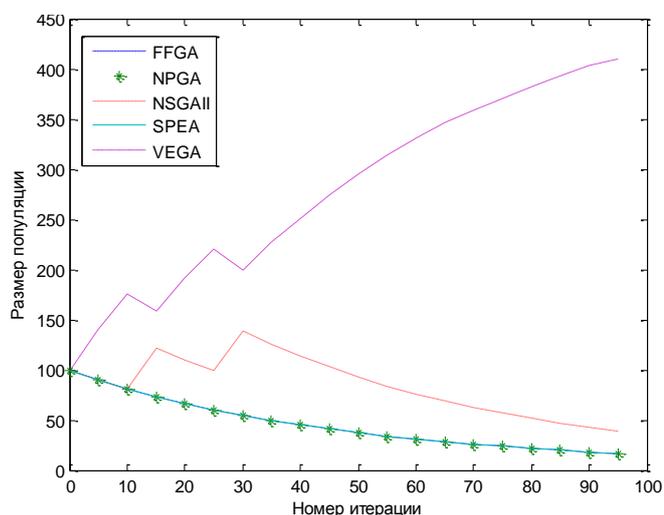
Было проведено исследование эффективности SMOGA на 18 тестовых функциях. Результаты тестирования алгоритмов приведены в таблице 1. Тестирование алгоритма проводилось при следующих условиях. В коэволюцию включены 5 различных алгоритмов VEGA, FFGA, NPGA, NSGAII и SPEA, параметры которых выбирались случайно. Для базовых алгоритмов приведены оценки при лучших параметрах, найденных при индивидуальном тестировании. На каждой задаче для оценки эффективности проводились 100 независимых запусков алгоритмов. Частные результаты усреднялись. Статистика обрабатывалась с помощью метода дисперсионного анализа - непараметрической оценки Манна-Уитни. Различия результатов, представленных в таблице 1, статистически значимы. В качестве критериев эффективности работы алгоритма выбраны следующие метрики: *generational distance (GD)*, *maximum spread (MS)*, *spacing (S)*.

Таблица 1 - Результаты тестирования алгоритмов на задачах, для которых известен истинный фронт Парето

Задача	Алгоритм	SPEA	VEGA	FFGA	NPGA	NSGAI	COMOGA	MEAN
	Критерий							
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
F5	GD	0,339	0,888	1,753	0,833	0,196	0,240	0,802
	MS	0,850	0,755	1,919	0,730	0,887	0,870	1,028
	S	0,475	0,251	0,094	0,228	0,346	0,908	0,279
F8	GD	0,821	1,411	4,221	1,709	0,650	0,394	1,762
	MS	0,754	1,170	3,655	0,859	0,852	0,786	1,458
	S	0,383	0,266	0,109	0,330	0,192	1,488	0,256
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

Можно отметить, что на множестве задач в среднем эффективность SelfCOMOGA может оказаться хуже лучшего из индивидуальных алгоритмов, но всегда выше средней эффективности по индивидуальным алгоритмам. При этом данный алгоритм не требует тщательной настройки параметров, что свойственно отдельно взятым алгоритмам.

Рисунок 1 - Динамика коэволюции для сложной задачи



применения предложенного подхода для автоматизирования проектирования интеллектуальных информационных технологий в задачах интеллектуального анализа данных, в которых равномерность покрытия множества Парето обеспечивает разнообразие представлений моделей знаний.

На рисунке 1 представлен пример динамики перераспределения ресурсов для условно сложной задачи.

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

В дальнейшем авторы планируют расширить область