УДК 519.8
DOI: 10.26467/2079-0619-2019-22-3-67-78

МОДИФИКАЦИЯ МЕТАЭВРИСТИЧЕСКОГО МЕТОДА ФЕЙЕРВЕРКОВ ДЛЯ ЗАДАЧ МНОГОКРИТЕРИАЛЬНОЙ ОПТИМИЗАЦИИ НА ОСНОВЕ НЕДОМИНИРУЕМОЙ СОРТИРОВКИ

А.В. ПАНТЕЛЕЕВ[1], А.Ю. КРЮЧКОВ[1]
[1]*Московский авиационный институт (национальный исследовательский университет), г. Москва, Россия*

В работе предлагается модификация численного метода фейерверков однокритериальной оптимизации для решения задач многокритериальной оптимизации. Метод относится к метаэвристическим алгоритмам, он не гарантирует нахождения точного решения, но может найти достаточно хорошее приближенное решение. Рассматриваются многокритериальные задачи оптимизации с числовыми критериями, имеющими одинаковую важность. Допустимое решение задачи представляется вектором из действительных чисел, значение каждой компоненты которого принадлежит определенному отрезку. Под оптимальным решением понимается решение, оптимальное по Парето. Так как точных решений, оптимальных по Парето, может быть бесконечно много, рассматривается способ нахождения приближения, состоящего из конечного числа решений, оптимальных по Парето. Модификация основана на процедуре недоминируемой сортировки, которая является основной процедурой для управления процессом поиска приближенного решения. Недоминируемая сортировка – это ранжирование решений на основе значений компонент числового вектора, полученных с помощью вычисления критериев. Каждая компонента соответствует определенному критерию, а множество решений разбивается на непересекающиеся подмножества. Первое подмножество – это решения, оптимальные по Парето, второе подмножество – это решения, оптимальные по Парето, если не учитывать первое подмножество, последнее подмножество – это решения, оптимальные по Парето, если не учитывать все предыдущие подмножества. После такого разбиения принимается решение о генерировании новых допустимых решений. Работа метода протестирована на общеизвестных задачах многокритериальной оптимизации с двумя критериями: ZDT2, LZ01. Задачи отличаются структурой расположения решений, оптимальных по Парето. Так LZ01 имеет достаточно сложную структуру решений, оптимальных по Парето. В заключении обсуждаются вопросы о дальнейшем направлении исследований и о возможности модификации метода для задач многокритериальной оптимизации с произвольными, а не параллелепипедными ограничениями.

Ключевые слова: многокритериальная оптимизация, метаэвристические методы, недоминируемая сортировка, оптимальность по Парето, теория принятия решений.

ВВЕДЕНИЕ

В современном мире проектирование новых технических систем становится все более сложным. Требования к системам в различных областях растет, поэтому приходится учитывать множество различных факторов [1, 2]. Эти факторы могут отражать противоположные цели, например, уменьшение энергопотребления и увеличение мощности процессора компьютера, уменьшение расходов и увеличение качества в процессе производства товаров. Важно найти компромисс между противоречащими друг другу требованиями. Как правило, для выбора оптимальных параметров формулируется задача оптимизации. Сравнение решений происходит на основе значений критериев, каждый критерий должен соответствовать определенному фактору и отражать степень влияния решения на этот фактор. Например, чем меньше значение критерия, тем больше экономия потребления энергии. Конечно, можно попытаться отразить влияние решения на различные факторы в одном критерии. Это упрощает задачу оптимизации, так как она становится задачей однокритериальной оптимизации. Для ее решения разработано множество методов, но здесь возникает сложность: как правило, нельзя описать влияние решения на все факторы в рамках одного критерия, поскольку можно потерять часть информации о связях между факторами, описывающими разные цели. Требуется разработка методов для решения задач многокритериальной оптимизации, позволяющих не преобразовывать несколько критериев в один.

Для соблюдения компромисса между критериями под оптимальным решением понимается решение, оптимальное по Парето. Нахождение таких решений может быть важным шагом как в дальнейшем процессе принятия решений, так и в получении всех возможных конфигураций системы, которые не позволяют улучшить значение по одному критерию без ухудшения значений по другим. Поэтому так важно разрабатывать новые эффективные методы для решения задач многокритериальной оптимизации.

Так как точных решений может быть бесконечно много и в общем случае нельзя найти способа описания решений, оптимальных по Парето, то предлагается рассмотреть способ нахождения приближенного решения. Приближенное решение представляет собой набор из конечного множества решений, близких к точному решению.

Можно выделить два направления разработки методов решения поставленной задачи: первое направление основано на аппроксимации оболочки Эджворта – Парето [3], а второе на аппроксимации границы Парето [4]. В статье рассматривается подход на основе аппроксимации границы Парето. Он связан с разработкой модификации метода фейерверков однокритериальной оптимизации [5], применимой для решения задач многокритериальной оптимизации. Метод относится к метаэвристическим алгоритмам, которые оказались эффективными при решении различных прикладных задач [6, 7].

ПОСТАНОВКА ЗАДАЧИ

Рассматривается задача многокритериальной оптимизации с параллелепипедными ограничениями. Предполагается, что все критерии имеют одинаковую важность и уменьшение значения одного критерия при фиксированных значениях остальных критериев более предпочтительно:

$$F(x)=\begin{pmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{pmatrix} \to \min_{x\in D}, \tag{1}$$

где $m \geq 2$ – число критериев, $D=\left\{x\in\mathbb{R}^n \middle| a_i \leq x_i \leq b_i, i=1,\dots,n\right\}$, $f_j : D \to \mathbb{R}$, $j=1,\dots,m.$

Требуется найти аппроксимацию множества допустимых решений, оптимальных по Парето. Для того чтобы дать определение решений, оптимальных по Парето, необходимо ввести несколько дополнительных определений.

Определение 1. Вектор $F(x)\in\mathbb{R}^m : F(x)=\left(f_1(x),\dots,f_m(x)\right)^T$ называется векторной оценкой решения $x\in D.$

Определение 2. Пусть $F^1=F\left(x^1\right), F^2=F\left(x^2\right)$ – векторные оценки решений $x^1\in D,\ x^2\in D.$ Оценка F^1 доминирует F^2 $(F^1 \prec F^2)$, если $\forall i\in\{1,\dots,m\}, F_i^1 \leq F_i^2$ и $\exists j\in\{1,\dots,m\} : F_j^1 < F_j^2.$

Определение 3. Решение $x_1 \in D$ предпочтительнее решения $x_2\in D : x_1 \prec x_2 \Leftrightarrow F(x_1) \prec F(x_2).$

Определение 4. $P=\left\{x\in D \middle| \nexists x'\in D : F(x') \prec F(x)\right\}$ является множеством решений, оптимальных по Парето.

Определение 5. Множество $PF=\left\{F(x) \middle| x\in P\right\}$ называется границей Парето.

Приближенным решением задачи (1) будет конечное множество решений, в котором каждый элемент достаточно близко расположен к какому-то элементу из P.

СТРАТЕГИЯ ПОИСКА РЕШЕНИЯ

Для решения задачи (1) будет использоваться модификация метода фейерверков [5], основанного на имитации процесса, происходящего во время фейерверка (салюта). Фейерверк сопровождается облаком светящихся осколков, заполняющих окрестность взорвавшегося заряда. В задачах оптимизации этот процесс ассоциируется с процедурой локального поиска.

Каждый залп салюта определяет переход от одной итерации поиска к другой (от одного поколения решений к другому). Сначала для реализации первого залпа определяются NP точек (решений) в множестве допустимых решений. В этих точках происходит взрыв, генерирующий определенное количество осколков, разлетающихся от точек взрыва в окрестности некоторого радиуса, определяемого для каждой точки в отдельности.

Далее идет процесс формирования нового поколения решений. Проводится недоминируемая сортировка. Недоминируемая сортировка – это ранжирование решений на основе их векторных оценок. Пусть $I = \left\{ x^p \middle| x^p \in D, p = 1,\ldots,NP \right\}$ – множество решений на текущей итерации, где $NP = |I| \ge 1$. Результатом сортировки является разбиение множества I на k непересекающихся подмножеств $Q_i, i = 1,\ldots,k, 1 \le k \le |I|$, где k – номер последнего подмножества в разбиении:

$$
\begin{aligned}
& I = \bigcup_{i=1}^{k} Q_i, \quad Q_i \bigcap Q_j = \varnothing, \quad i \ne j, \\
& Q_1 = \left\{ x \in I \middle| \nexists\, x' \in I : F(x') \prec F(x) \right\}, \\
& \vdots \\
& Q_l = \left\{ x \in I \setminus \bigcup_{i=1}^{l-1} Q_i \middle| \nexists\, x' \in I \setminus \bigcup_{i=1}^{l-1} Q_i : F(x') \prec F(x) \right\}, \\
& \vdots \\
& Q_k = I \setminus \bigcup_{i=1}^{k-1} Q_i .
\end{aligned}
$$

Другими словами, недоминируемая сортировка представляет собой повторяющуюся процедуру выделения предпочтительных решений. На первом шаге выбираются предпочтительные решения из I. Далее эти предпочтительные решения удаляются из I, и процедура повторяется к оставшейся части.

Среди решений, соответствующих точкам взрыва и полученным осколкам, выбираются решения с недоминируемыми векторными оценками (множество Q_1). Остальные решения выбираются из оставшихся случайным образом с вероятностью, определяемой расстоянием в пространстве критериев до других точек (чем больше суммарное расстояние, тем больше вероятность выбора).

Процесс поиска завершается при достижении заданного числа итераций.

История развития процедуры недоминируемой сортировки связана с историей развития численных методов для решения многокритериальных задач оптимизации. Процедура недоминируемой сортировки использовалась в различных алгоритмах многокритериальной оптимизации. Основные усилия были направлены на уменьшение алгоритмической сложности процедуры. В итоге в работе [8] был предложен алгоритм, имеющий сложность $O(n \log^{m-1} n)$, где n – число решений на текущей итерации, m – число критериев. Позже в [9] была предложена модификация алгоритма с такой же оценкой сложности, но в худшем случае.

АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ

Далее описаны шаги работы модифицированного метода «фейерверков».

Шаг 1. Задать параметры метода: число зарядов на каждой итерации $NP \in \mathbb{N}$; параметр $m > 0$, контролирующий число осколков; $s_{\min}, s_{\max} \in \mathbb{N}, s_{\min} \le s_{\max}$ – минимальное и максимальное число осколков для каждого заряда; максимальная амплитуда взрыва $A_{\max} > 0$; максимальное число итераций $Iter_{\max} \in \mathbb{N}$; номер итерации, после которой используется новая стратегия отбора решений $I_b \in \{0\} \bigcup \mathbb{N}, I_b \le I_{\max}$.

Шаг 2. Положить $iter = 1$ (счетчик числа итераций). Генерировать NP решений на множестве допустимых решений D, $I^{iter} = \left\{x^{1,1}, \ldots, x^{NP,1}\right\}$:

$$x_i^{p,1} = a_i + Urand(0;1) \cdot \left(b_i - a_i\right),$$

где $i = 1, \ldots, n, p = 1, \ldots, NP, Urand\left(0;1\right)$ – случайная величина, имеющая равномерное распределение на отрезке $\left[0;1\right]$.

Шаг 3. Провести недоминируемую сортировку множества I^{iter}. Оно разбивается на $1 \le l \le \left|I^{iter}\right|$ подмножеств Q_i: $I^{iter} = \bigcup_{i=1}^{l} Q_i$.

Шаг 4. Процедура взрыва и генерации осколков.

Шаг 4.1. Для всех $p = 1, \ldots, NP$ вычислить:

1. Номер подмножества $q : x^{p,iter} \in Q_q, 1 \le q \le l$.
2. Число осколков:

$$s^{p,iter} = m \cdot \log_2 \left(1 + \frac{l}{q}\right) \cdot \left(1 - \frac{\left|Q_q\right|}{NP}\right), \quad \hat{s}^{p,iter} = \begin{cases} s_{\min}, \left[s^{p,iter}\right] \le s_{\min}, \\ s_{\max}, \left[s^{p,iter}\right] \ge s_{\max}, \\ \left[s^{p,iter}\right], \text{в остальных случаях,} \end{cases}$$

где l – номер последнего подмножества в разбиении, $\hat{s}^{p,iter}$ – количество осколков, порождаемых взрывом в точке $x^{p,iter}$.

Шаг 4.2. Определение положения осколков. Для каждого $p = 1, \ldots, NP$ найти положения осколков с номерами $s = 1, \ldots, \hat{s}^{p,iter}$:

1. Найти $q : x^{p,iter} \in Q_q, 1 \le q \le l$.
2. Положить $\tilde{x}^{p,iter,s} = x^{p,iter}$.
3. Для каждого номера осколка s:
   3.1. Положить $\xi = Urand(0;1)$.
   3.2. Найти число исследуемых координатных направлений:

$$\hat{n} = \left[n \cdot \xi\right],$$

где $[\cdot]$ – целая часть числа.

4. Если сгенерированная случайная величина $\xi < 0,5,$ то применить первый способ определения положения осколков:

4.1. Выбрать случайным образом $\hat{n}$ координат для каждого номера осколка s.

4.2. Вычислить амплитуду (радиус) взрыва для всех $p = 1,\dots,NP$:

$$A^{p,iter} = A_{\max} \cdot \log_2\left(1 + \frac{q}{l}\right) \cdot \frac{|Q_q|}{NP}.$$

4.3. Для каждого выбранного номера i из $\hat{n}$ координат вектора $\tilde{x}^{p,iter,s}$ и номера осколка s вычислить приращение:

$$h_i^s = A^{p,iter} \cdot Urand(-1;1),$$
$$\tilde{x}_i^{p,iter,s} = x_i^{p,iter} + h_i^s.$$

5. Если сгенерированная случайная величина $\xi \geq 0,5,$ то применить второй способ определения положения осколков:

5.1. Выбрать случайным образом $\hat{n}$ координат для каждого номера осколка s.

5.2. Для каждого выбранного номера i из $\hat{n}$ координат вектора $\tilde{x}^{p,iter,s}$ и номера осколка s вычислить приращение:

$$\tilde{x}_i^{p,iter} = x_i^{p,iter} \cdot Nrand(1;1),$$

где $Nrand(1;1)$ – случайная величина, имеющая нормальное распределение с математическим ожиданием, равным 1, и дисперсией, равной 1.

Шаг 4.3. Проверка выхода за границу множества допустимых решений D.

1. Для каждого $p = 1,\dots,NP$ проверить:

1.1. Для каждого $s = 1,\dots,\hat{s}^{p,iter}$ проверить:

Если $\tilde{x}_i^{p,iter,s} \notin [a_i;b_i]$, то

$$\tilde{x}_i^{p,iter,s} = \begin{cases} Urand\left(a_i;0,5\cdot(a_i+b_i)\right), & \tilde{x}_i^{p,iter,s} < a_i, \\ Urand\left(0,5\cdot(a_i+b_i);b_i\right), & \tilde{x}_i^{p,iter,s} > b_i, \end{cases}$$

где $i = 1,\dots,n$.

1.2. Добавить $\tilde{x}_i^{p,iter,s}$ к $I^{iter}: I^{iter} = I^{iter} \bigcup \left\{\tilde{x}_i^{p,iter,s}\right\}$.

Шаг 5. Создание новых решений.

Шаг 5.1. Провести недоминируемую сортировку $I^{iter} = \bigcup_{i=1}^{l} Q_i$, l – номер последнего подмножества в разбиении I^{iter}. Увеличить счетчик числа итераций: $iter = iter + 1,$ $I^{iter} = \varnothing.$

Шаг 5.2. Если $I_b \leq iter - 1,$ то выполнить шаг 5.3, иначе шаг 5.4.

Шаг 5.3. Если $|Q_1| \ge NP,$ то $P = Q_1.$ Если $|Q_1| < NP,$ то $I^{iter} = Q_1, P = \bigcup_{i=2}^{l} Q_i.$ Перейти к шагу 5.5.

Шаг 5.4. Найти $u_{\min}$:

$$u_{\min} = \min_{1 \le u \le l} \left\{ u : \left| \bigcup_{i=1}^{u} Q_i \right| \ge NP \right\}.$$

Если $u_{\min} = 1,$ то $P = Q_1.$ Если $u_{\min} \ne 1$ и $\left| \bigcup_{i=1}^{u_{\min}} Q_i \right| = NP$, то $I^{iter} = \bigcup_{i=1}^{u_{\min}} Q_i, P = \varnothing,$ иначе $I^{iter} = \bigcup_{i=1}^{u_{\min}-1} Q_i, P = Q_{u_{\min}}.$

Шаг 5.5. Если $P \ne \varnothing$, то для каждой точки $x^w \in P$ подсчитать $R\left(x^w\right)$ – сумму расстояний до остальных точек, $p\left(x^w\right)$ – вероятность взрыва:

$$R\left(x^w\right) = \sum_{x^b \in P} \rho\left(F\left(x^w\right), F\left(x^b\right)\right), \quad p\left(x^w\right) = \frac{R\left(x^w\right)}{\sum_{x^b \in P} R\left(x^q\right)},$$

где $\rho(x, y)$ – евклидово расстояние между векторами $x, y \in \mathbb{R}^n$.

Шаг 5.6. Используя вероятность $p\left(x^w\right)$, случайным образом выбрать из множеств P множество точек (решений) в количестве $NP - \left|I^{iter}\right|$ и добавить их в множество I^{iter}.

Шаг 5.7. Если $iter \le Iter_{\max}$, то перейти к шагу 3. Иначе в качестве приближенного решения взять I^{iter}.

Стоит заметить, что проводить недоминируемую сортировку на шаге 3 после первой итерации необязательно. Информацию о разбиении I^{iter} можно взять после выполнения шага 5.1.

ТЕСТИРОВАНИЕ РАБОТЫ МЕТОДА

Основной интерес для измерения представляют два свойства решения: как близко оно расположено к истинной границе Парето и насколько равномерно ее покрытие. Для измерения этих свойств была выбрана метрика, которая называется гиперобъем [10]. Гиперобъем $HV(I, r)$ зависит от решения I и точки r в пространстве критериев. Он определяется следующим образом:

$$HV(I, r) = \mu\left(\bigcup_{x \in I} \left[r_1; F_1(x)\right] \times \ldots \times \left[r_m; F_m(x)\right]\right),$$

где $\mu(\cdot)$ – мера Лебега.

Далее приведены результаты 10 запусков решения задач с разными параметрами. После каждого запуска было получено приближенное решение I_i. Оно имело размер NP_i, где NP_i – число зарядов при i-м запуске. В результате были получены значения $\{HV_i : HV_i = HV(I_i, r), i = 1, \ldots, 10\}$. Во всех тестах $r = (0;0)$. Аналогичным образом были рассчитаны значения гиперобъема $\widetilde{HV_i}$ для истинного фронта Парето (так как задачи тестовые, то для них известно множество решений, оптимальных по Парето). Число точек, по которым считался гиперобъем для точного решения, также равнялся NP_i. В таблицах приведены следующие характеристики: среднее значение $\overline{HV} = \sum_{i=1}^{10}\left(HV_i - \widetilde{HV_i}\right)/10$, максимальные и минимальные значения $HV_{\max} = \max_{i=1,\ldots,10}\left\{HV_i - \widetilde{HV_i}\right\}$, $HV_{\min} = \min_{i=1,\ldots,10}\left\{HV_i - \widetilde{HV_i}\right\}$.

ТЕСТОВЫЕ ЗАДАЧИ

Первая задача ZDT2 [11] имеет выпуклую границу Парето. Она имеет следующий вид:

$$f_1(x) = x_1,$$

$$f_2(x) = g(x) \cdot \left(1 - \left(\frac{x_1}{g(x)}\right)^2\right),$$

где $g(x) = 1 + 9\sum_{i=2}^{n} x_i / (n-1), n \geq 2, x_i \in [0;1], i = 1, \ldots, n.$ Во всех тестах $n = 30$.

Множество векторных оценок найденного решения представлено на рис. 1. PF здесь и далее обозначает множество эффективных векторных оценок. Параметры и значения метрики представлены в табл. 1.

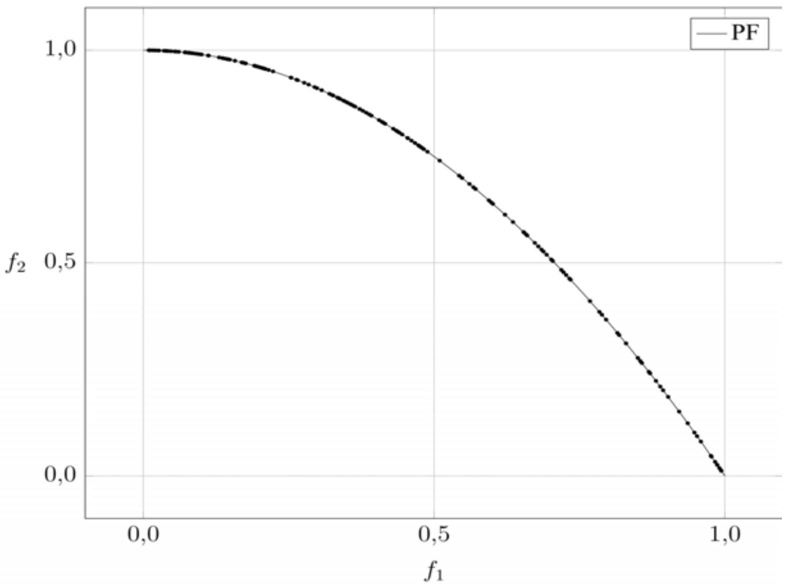

Рис. 1. Множество векторных оценок найденного решения для задачи ZDT2
Fig. 1. A set of vectors' values for the ZDT2 problem solution

Таблица 1
Table 1

Параметры метода и значения метрики для задачи ZDT2
Parameters of method and metric values for the ZDT2 problem

I_{max}	NP	m	A_{max}	s_{min}	s_{max}	I_b	$\overline{HV}$	HV_{min}	HV_{max}
200	200	10	1,1	5	20	0	6,3012E-003	3,4064E-003	1,0032E-002
250	200	20	1,2	10	30	0	6,7858E-003	3,6735E-003	1,2382E-002
300	200	25	1,5	5	30	0	6,2619E-003	3,5421E-003	1,1649E-002
600	200	15	1,05	20	50	0	7,6412E-003	3,9464E-003	2,0607E-002
600	200	20	0,9	20	50	0	8,7471E-003	5,9410E-003	1,6159E-002

Вторая задача LZ01 [12] имеет сложную структуру решений, оптимальных по Парето. В оригинальной статье задача называется F_1. Она имеет следующий вид:

$$f_1(x) = x_1 + \frac{2}{|J_1|}\sum_{j\in J_1}\left(x_j - x_1^{0,5\cdot\left(1+\frac{3\cdot(j-1)}{n-2}\right)}\right)^2,$$

$$f_2(x) = 1-\sqrt{x_1} + \frac{2}{|J_2|}\sum_{j\in J_2}\left(x_j - x_1^{0,5\cdot\left(1+\frac{3\cdot(j-1)}{n-2}\right)}\right)^2,$$

где $J_1 = \{j : j \text{ нечётное}, 2 \le j \le n\}$, $J_2 = \{j : j \text{ чётное}, 2 \le j \le n\}$, $n \ge 2$, $x_i \in [0;1]$, $i = 1,\dots,n.$ Во всех тестах $n = 30$.

Множество векторных оценок найденного решения представлено на рис. 2. Параметры и значения метрики представлены в табл. 2.

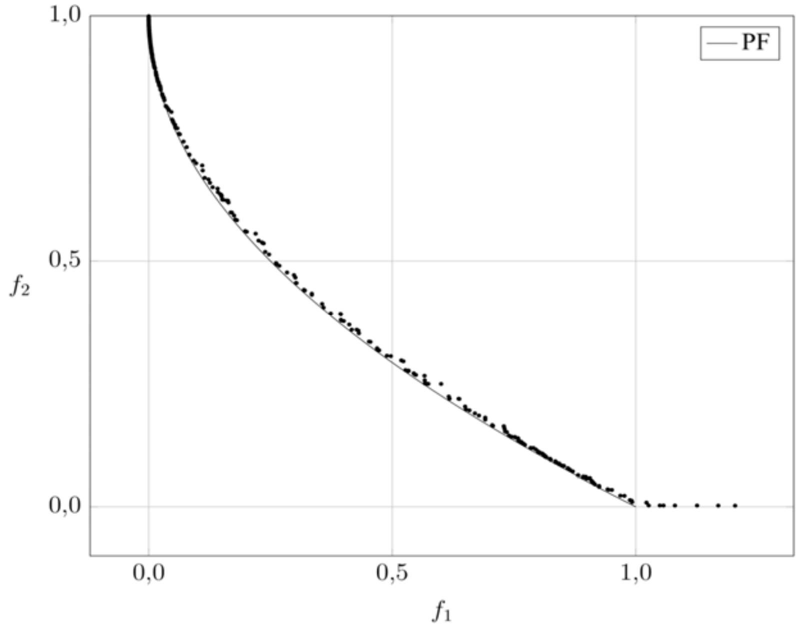

Рис. 2. Множество векторных оценок найденного решения для задачи LZ01
Fig. 2. A set of vectors' values for the LZ01 problem solution

Таблица 2
Table 2

Параметры метода и значения метрики для задачи LZ01
Parameters of method and metric values for the LZ01 problem

$I_{\max}$	NP	m	$A_{\max}$	$s_{\min}$	$s_{\max}$	I_b	$\overline{HV}$	$HV_{\min}$	$HV_{\max}$
500	400	10	1,1	8	15	0	8,3839E-03	6,9525E-03	1,0721E-02
500	400	20	0,7	8	15	0	8,0824E-03	6,0186E-03	9,6019E-03
500	400	25	1,5	8	15	0	8,8832E-03	6,7740E-03	1,0672E-02
500	400	15	1,05	8	15	0	8,5516E-03	6,0967E-03	1,0432E-02
500	400	20	0,5	8	15	0	8,0425E-03	7,0934E-03	1,0182E-02

ОБСУЖДЕНИЕ ПОЛУЧЕННЫХ РЕЗУЛЬТАТОВ И ЗАКЛЮЧЕНИЕ

Основные проблемы, которые нужно решить при создании метода для решения задач многокритериальной оптимизации, связаны с обеспечением сходимости к истинной границе и равномерности ее покрытия. Процедура недоминируемой сортировки не решает этих проблем. Она всего лишь позволяет ранжировать решения. Вопрос о правилах формирования новых решений остается открытым для исследователей. Недоминируемая сортировка основана на принципе, который заложен в определении оптимальности по Парето, поэтому является хорошей эвристикой. У этой процедуры существует недостаток. Она позволяет определить, является ли одно решение предпочтительнее другого, но не позволяет определить количественную меру этого предпочтения. Подход на основе недоминируемой сортировки не является единственным. Существуют и другие эвристики для выделения решений [13].

Дальнейшие исследования необходимо продолжить в направлении улучшения механизмов отбора решений для обеспечения более равномерного покрытия истинной границы Парето при меньшем числе разрядов. Для решения задач многокритериальной оптимизации с произвольными ограничениями, задающими множество допустимых решений, необходимо модифицировать процедуру недоминируемой сортировки. При ранжировании решений уже недостаточно просто сравнивать их на основе доминирования. Необходимо учитывать, лежит ли решение в множестве допустимых решений и насколько сильно оно выходит за допустимые границы. Из-за этого правила разбиения на подмножества необходимо поменять. Например, если два решения являются оптимальными по Парето, но первое решение лежит в множестве допустимых решений, а второе нет, то первое доминирует второе. Основная идея заключается в пересмотре понятия доминирования. Существует успешный пример применения такого подхода к задачам многокритериальной оптимизации с ограничениями произвольного типа [14].

СПИСОК ЛИТЕРАТУРЫ

1. **Arias-Montano A., Coello Coello A.C., Mezura-Montes E.** Multiobjective evolutionary algorithms in aeronautical and aerospace engineering // IEEE Transactions on Evolutionary Computation. 2012. Vol. 16, Iss. 5. Pp. 662–694. DOI: 10.1109/TEVC.2011.2169968

2. Multi-objective optimization. Techniques and applications in chemical engineering / Ed. G.P. Rangaiah. 2nd ed. World Scientific, 2017. 588 p. DOI: 10.1142/10240

3. **Березкин В.Е., Лотов А.В., Лотова Е.А.** Изучение гибридных методов аппроксимации оболочки Эджворта – Парето в нелинейных задачах многокритериальной оптимизации // Журнал вычислительной математики и математической физики. 2014. Т. 54, № 6. С. 905–918. DOI: 10.7868/S0044466914060039

4. Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач. М.: Наука, 1982. 256 с.

5. Tan Y., Zhu Y. Fireworks Algorithm for Optimization // Advances in Swarm Intelligence. First International Conference. ICSI 2010, Beijing, China, June 12–15, 2010. Proceedings, Part I / Ed. Y. Tan, Y. Shi, K.C. Tan. Springer, Berlin, Heidelberg, 2010. Pp. 355–364. Lecture Notes in Computer Science, vol. 6145. DOI: 10.1007/978-3-642-13495-1_44

6. Handbook of Metaheuristics / Ed. F. Glover, G.A. Kochenberger. New-York; Boston; Moscow: Kluwer Academic Publishers, 2003. 557 p.

7. Пантелеев А.В., Крючков А.Ю. Метаэвристические методы оптимизации в задачах оценки параметров динамических систем // Научный Вестник МГТУ ГА. 2017. Т. 20, № 2. С. 37–45. DOI: 10.26467/2079-0619-2017-20-2-37-45

8. Fortin F.-A., Grenier S., Parizeau M. Generalizing the improved run-time complexity algorithm for non-dominated sorting // GECCO '13. Proceedings of the 15th annual conference on Genetic and evolutionary computation. Amsterdam, The Netherlands – July 06–10, 2013. 2013. Pp. 615–622. DOI: 10.1145/2463372.2463454

9. Buzdalov M., Shalyto A. A Provably Asymptotically Fast Version of the Generalized Jensen Algorithm for Non-dominated Sorting // Parallel Problem Solving from Nature – PPSN XIII: 13th International Conference, Ljubljana, Slovenia, September 13–17, 2014: proceedings. Cham: Springer International Publishing, 2014. Pp. 528–537. DOI: 10.1007/978-3-319-10762-2_52

10. Zitzler E., Thiele L. Multiobjective optimization using evolutionary algorithms – a comparative case study // Parallel Problem Solving from Nature – PPSN V. 5th International Conference Amsterdam, The Netherlands September 27–30, 1998. 1998. Pp. 292–301. DOI: 10.1007/BFb0056872

11. Zitzler E., Deb K., Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results // Evolutionary Computation. 2000. Vol. 8, № 2. Pp. 173–195. DOI: 10.1162/106365600568202

12. Li H., Zhang Q. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II // IEEE Transactions on Evolutionary Computation. 2009. Vol. 13, Iss. 2, April. Pp. 284–302. DOI: 10.1109/TEVC.2008.925798

13. Amuso V.J., Enslin J. The Strength Pareto Evolutionary Algorithm 2 (SPEA2) applied to simultaneous multi-mission waveform design // 2007 International waveform diversity and design conference. Pisa, 2007. Pp. 407–417. DOI: 10.1109/WDDC.2007.4339452

14. Jain H., Deb K. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach. Part II: Handling constraints and extending to an adaptive approach // IEEE Transactions on evolutionary computation. 2014. Vol. 18, Iss. 4, Aug. Pp. 602–622. DOI: 10.1109/TEVC.2013.2281534

СВЕДЕНИЯ ОБ АВТОРАХ

Пантелеев Андрей Владимирович, доктор физико-математических наук, профессор, заведующий кафедрой математической кибернетики Московского авиационного института (национального исследовательского университета), avpanteleev@inbox.ru.

Крючков Александр Юрьевич, магистрант Московского авиационного института (национального исследовательского университета), alex9x99@yandex.ru.

MODIFICATION OF FIREWORKS METHOD FOR MULTIOBJECTIVE OPTIMIZATION BASED ON NON-DOMINATED SORTING

Andrei V. Panteleev[1], Alexander U. Krychkov[1]
[1]Moscow Aviation Institute (National Research University), Moscow, Russia

ABSTARCT

The article suggests a modification for numerical fireworks method of the single-objective optimization for solving the problem of multiobjective optimization. The method is metaheuristic. It does not guarantee finding the exact solution, but can give a good approximate result. Multiobjective optimization problem is considered with numerical criteria of equal importance. A possible solution to the problem is a vector of real numbers. Each component of the vector of a possible solution belongs to a certain segment. The optimal solution of the problem is considered a Pareto optimal solution. Because the set of Pareto optimal solutions can be infinite; we consider a method for finding an approximation consisting of a finite number of Pareto optimal solutions. The modification is based on the procedure of non-dominated sorting. It is the main procedure for solutions search. Non-dominated sorting is the ranking of decisions based on the values of the numerical vector obtained using the criteria. Solutions are divided into disjoint subsets. The first subset is the Pareto optimal solutions, the second subset is the Pareto optimal solutions if the first subset is not taken into account, and the last subset is the Pareto optimal solutions if the rest subsets are not taken into account. After such a partition, the decision is made to create new solutions. The method was tested on well-known bi-objective optimization problems: ZDT2, LZ01. Structure of the location of Pareto optimal solutions differs for the problems. LZ01 have complex structure of Pareto optimal solutions. In conclusion, the question of future research and the issue of modifying the method for problems with general constraints are discussed.

Key words: multiobjective optimization, metaheuristic algorithms, non-dominated sorting, Pareto efficiency, Pareto optimality, decision making.

REFERENCES

1. **Arias-Montano, A., Coello Coello, A.C. and Mezura-Montes, E.** (2012). *Multiobjective evolutionary algorithms in aeronautical and aerospace engineering.* IEEE Transactions on Evolutionary Computation, vol. 16, iss. 5, pp. 662–694. DOI: 10.1109/TEVC.2011.2169968
2. *Multi-objective optimization. Techniques and applications in chemical engineering* (2017). Ed. G.P. Rangaiah. 2nd ed. World Scientific, 588 p. DOI: 10.1142/10240
3. **Berezkin, V.E., Lotov, A.V. and Lotova, E.A.** (2014). *Study of hybrid methods for approximating the Edgeworth-Pareto hull in nonlinear multicriteria optimization problems*. Computational mathematics and mathematical physics, vol. 54, no. 6, pp. 919–930. DOI: 10.7868/S0044466914060039
4. **Podinovskij, V.V. and Nogin, V.D.** (1982). *Pareto-optimalnyye resheniya mnogokriterialnkyh zadach* [Pareto optimal solutions of multiobjective problems]. Moscow: Nauka, 256 p. (in Russian)
5. **Tan, Y. and Zhu, Y.** (2010). *Fireworks algorithm for optimization.* In: Tan Y., Shi Y., Tan K.C. (eds.). Advances in Swarm Intelligence. First International Conference, ICSI 2010, Beijing, China, June 12–15, 2010, Proceedings, Part I. ICSI 2010. Lecture Notes in Computer Science, vol. 6145. Berlin, Heidelberg: Springer, pp. 355–364. DOI: 10.1007/978-3-642-13495-1_44
6. *Handbook of Metaheuristics* (2003). Ed. F. Glover and G.A. Kochenberger. New-York; Boston; Moscow: Kluwer Academic Publishers, 557 p.
7. **Panteleev, A.V. and Krychkov, A.U.** (2017). *Metaheuristic optimization methods for parameters estimation of dynamical systems.* Civil Aviation High Technologies, vol. 20, no. 2, pp. 37–45. DOI: 10.26467/2079-0619-2017-20-2-37-45 (in Russian)
8. **Fortin F.-A., Grenier S. and Parizeau M.** (2013). *Generalizing the improved run-time complexity algorithm for non-dominated sorting.* GECCO '13. Proceedings of the 15th annual confer-

ence on Genetic and evolutionary computation. Amsterdam, The Netherlands – July 06–10, 2013, pp. 615–622. DOI: 10.1145/2463372.2463454

9. Buzdalov, M. and Shalyto, A. (2014). *A Provably Asymptotically Fast Version of the Generalized Jensen Algorithm for Non-dominated Sorting.* Parallel Problem Solving from Nature – PPSN XIII: 13th International Conference, Ljubljana, Slovenia, September 13–17, 2014. Proceedings. Cham: Springer International Publishing, pp. 528–537. DOI: 10.1007/978-3-319-10762-2_52

10. Zitzler, E. and Thiele, L. (1998). *Multiobjective optimization using evolutionary algorithms – a comparative case study.* Parallel Problem Solving from Nature – PPSN V. 5th International Conference Amsterdam, The Netherlands September 27–30, 1998, pp. 292–301. DOI: 10.1007/BFb0056872

11. Zitzler, E., Deb, K. and Thiele, L. (2000). *Comparison of multiobjective evolutionary algorithms: empirical results.* Evolutionary Computation, vol. 8, no. 2, pp. 173–195. DOI: 10.1162/106365600568202

12. Li, H. and Zhang, Q. (2009). *Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II.* IEEE Transactions on Evolutionary Computation, vol. 13, iss. 2, April, pp. 284–302. DOI: 10.1109/TEVC.2008.925798

13. Amuso, V.J. and Enslin, J. (2007). *The Strength Pareto Evolutionary Algorithm 2 (SPEA2) applied to simultaneous multi-mission waveform design.* 2007 International waveform diversity and design conference, Pisa, pp. 407–417. DOI: 10.1109/WDDC.2007.4339452

14. Jain, H. and Deb, K. (2014). *An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach. Part II: Handling constraints and extending to an adaptive approach.* IEEE Transactions on evolutionary computation, vol. 18, iss. 4, Aug., pp. 602–622. DOI: 10.1109/TEVC.2013.2281534

INFORMATION ABOUT THE AUTHORS

Andrei V. Panteleev, Doctor of Physical and Mathematical Sciences, Professor, Head of Mathematics and Cybernetics Chair, Moscow Aviation Institute (National Research University), avpanteleev@inbox.ru.

Alexander U. Krychkov, Master Degree Student of Mathematics and Cybernetics Chair, Moscow Aviation Institute (National Research University), alex9x99@yandex.ru.

Поступила в редакцию	25.11.2018	Received	25.11.2018
Принята в печать	21.05.2019	Accepted for publication	21.05.2019