

Исследование алгоритмов NSGA-III и AGE-MOEA-II для решения задач многокритериальной оптимизации

Э.М. Вихтенко, Д.В. Зубков

Тихоокеанский государственный университет, Хабаровск

Аннотация: Статья посвящена рассмотрению методов многокритериальной оптимизации по Парето, построенных на основе генетических алгоритмов. Рассмотрены методы NSGA-III и AGE-MOEA-II, дан их сравнительный анализ. Полученные результаты важны как для теоретических исследований в области генетических алгоритмов, так и для практического применения в инженерии и других сферах, где многокритериальная оптимизация играет ключевую роль.

Ключевые слова: задача многокритериальной оптимизации, фронт Парето, генетический алгоритм, NSGA-III, AGE-MOEA-II.

Введение

В последние десятилетия большую популярность у разработчиков информационных систем самого разного назначения получают алгоритмы, построенные как аналоги биологических процессов. К таким алгоритмам относятся и так называемые эволюционные вычисления, в частности, генетические алгоритмы. Генетические алгоритмы (ГА) применяются для решения задач оптимизации, область их применения постоянно расширяется. Достаточно традиционными сферами применения ГА являются различные задачи, связанные с составлением планов перевозок [1] или планированием действий [2, 3]. В то же время ГА становятся инструментом для исследования и развития других методов [4]. Широкое распространение ГА приводит к необходимости создания новых модификаций алгоритмов, обеспечивающих эффективное решение оптимизационных задач, что, в свою очередь, делает данный класс алгоритмов не инструментом, а объектом исследований [5, 6].

Генетические алгоритмы являются мощным инструментом для работы в больших и сложных пространствах поиска, поскольку они имитируют процессы естественного отбора и генетических изменений, позволяя

одновременно исследовать множество потенциальных решений. Это обеспечивает эффективный поиск по широкому диапазону возможных решений и способствует нахождению оптимальных ответов даже в условиях, где традиционные методы оптимизации могут быть неэффективны. Такая способность адаптироваться и исследовать различные варианты делает ГА эффективными решениями для сложных задач, где пространство поиска решений обширно и/или имеет сложную структуру.

Для решения задач оптимизации ГА используют закодированную последовательность параметров, представленную в виде набора чисел – так называемую «цепочку ДНК». Таким образом, этот класс алгоритмов представляет высокую гибкость и адаптивность, так как специфика прикладной задачи кодируется в ДНК, необходимости изменения самого алгоритма не возникает.

Так же важную роль в поиске подходящего решения (или набора решений) играет способ задания целевой функции (фитнесс-функции, функции приспособленности). Данная функция используется для оценки степени пригодности того или иного решения относительно других решений.

В случае решения многокритериальных задач оптимизации для оценки полученных решений используется так называемый фронт Парето [7-9]. Парето-фронт представляет из себя множество решений, каждое из которых удовлетворяет условию «оптимальности по Парето» [10, 11]. При решении задачи оптимизации по M параметрам (критериям) фронт Парето является множеством точек в пространстве R^M , каждая из которых представляет из себя решение, которое нельзя улучшить по одному критерию, не ухудшив по другому. С задачей построения Парето-фронта сталкивается покупатель, когда выбирает товар по критерию «оптимальные цена – качество».

В работе [12], опубликованной в 2015 году, выполнен сравнительный анализ генетических алгоритмов SPEA, SPEA2, VEGA, FFGA, NSGA,

использующих оптимальность по Парето. В данной работе рассмотрим пару алгоритмов NSGA-III и AGE-MOEA-II, каждый из которых использует недоминируемую сортировку популяции для построения Парето-фронт. Алгоритм NSGA-III (Non-dominated Sorting Genetic Algorithm) [13] разработан в 2012 году и успешно используется зарубежными исследователями для решения прикладных задач [14, 15]. Адаптивный алгоритм AGE-MOEA-II (Adaptive Geometry Estimation based Many Objectives Evolutionary Algorithm) опубликован в 2022 году [16]. В русскоязычном сегменте статьи, связанные с данными алгоритмами, практически отсутствуют. Так, например, в библиотеке eLibrary на запрос по AGE-MOEA-II находится только одна работа [17].

Прежде чем переходить к изложению полученных авторами результатов сравнительного анализа алгоритмов NSGA-III и AGE-MOEA-II, приведем краткое их описание.

Алгоритмы NSGA-III и AGE-MOEA-II

Как всякий генетический алгоритм, NSGA-III состоит из выполнения подготовительного этапа – формирования начальной популяции, а затем основного этапа – циклов по развитию популяции, получению новых поколений. Считаем, что выполнено t циклов алгоритма, т.е. сформировано поколение P_t .

Псевдокод алгоритма NSGA-III из оригинальной статьи представлен на рисунке 1.

Цикл формирования $(t+1)$ -го поколения состоит из следующих шагов:

1) выполняются операции скрещивания и мутации, в результате которых генерируется поколение потомков P_{t+1} на основе родительской популяции P_t ;

2) проводится недоминируемая сортировка популяций P_t и P_{t+1} . На этом шаге производится анализ решений и выборка тех решений, для

которых выполняется критерий эффективности по Парето – ни один из критериев не может быть улучшен без ухудшения другого критерия;

Algorithm 1 Generation t of NSGA-III procedure

Input: H structured reference points Z^s or supplied aspiration points Z^a , parent population P_t

Output: P_{t+1}

- 1: $S_t = \emptyset, i = 1$
- 2: $Q_t = \text{Recombination+Mutation}(P_t)$
- 3: $R_t = P_t \cup Q_t$
- 4: $(F_1, F_2, \dots) = \text{Non-dominated-sort}(R_t)$
- 5: **repeat**
- 6: $S_t = S_t \cup F_i$ and $i = i + 1$
- 7: **until** $|S_t| \geq N$
- 8: Last front to be included: $F_l = F_i$
- 9: **if** $|S_t| = N$ **then**
- 10: $P_{t+1} = S_t$, break
- 11: **else**
- 12: $P_{t+1} = \cup_{j=1}^{l-1} F_j$
- 13: Points to be chosen from F_l : $K = N - |P_{t+1}|$
- 14: Normalize objectives and create reference set Z^r :
 $\text{Normalize}(f^a, S_t, Z^r, Z^s, Z^a)$
- 15: Associate each member s of S_t with a reference point:
 $[\pi(s), d(s)] = \text{Associate}(S_t, Z^r)$ % $\pi(s)$: closest reference point, d : distance between s and $\pi(s)$
- 16: Compute niche count of reference point $j \in Z^r$: $\rho_j = \sum_{s \in S_t/F_l} ((\pi(s) = j) ? 1 : 0)$
- 17: Choose K members one at a time from F_l to construct P_{t+1} : $\text{Niching}(K, \rho_j, \pi, d, Z^r, F_l, P_{t+1})$
- 18: **end if**

Рис. 1 – Псевдокод алгоритма NSGA-III

3) если количество особей (решений) равно размеру популяции, то считаем, что поколение P_{t+1} построено. В этом случае оно передается в основную программу для проверки условия завершения вычислений;

4) если количество особей отличается от размера популяции, то выполняются действия, записанные в строках 12-17 псевдокода (рис. 1):

4.1) проводится нормализация решений и создание опорного множества. Нормализация проводится для значений целевой функции по норме пространства L_p , в результате нормализации все значения располагаются в диапазоне $[0;1]$. Опорное множество получается проекцией

точек на гиперплоскость, проходящую через крайние точки фронта. Данный шаг позволяет значительно ускорить нахождения оптимальных решений;

4.2) нормализованные решения связываются («ассоциируются») с опорными точками из опорного множества (рис. 2). На данном шаге производится оценка расстояния от полученного решения до ближайшей опорной точки;

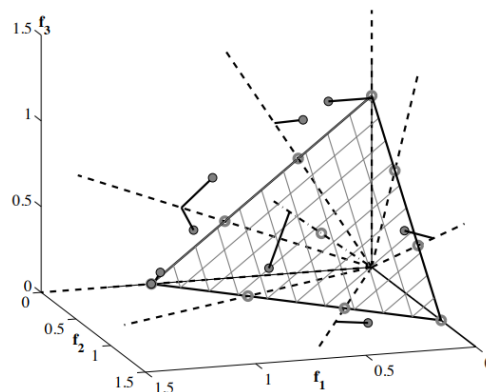


Рис. 2 – Ассоциация решений с опорными точками

4.3) выполняется добавление в поколение P_{t+1} решений, максимально близких к опорному множеству.

Детальное описание реализации шагов алгоритма можно найти в работах [13, 18, 19].

Алгоритм AGE-МОЕА-II так же строится на нахождении множества недоминируемых решений, описывающего в некотором смысле оптимальный Парето-фронт. В AGE-МОЕА-II для определения характеристик Парето-фронта использован итерационный метод Ньютона-Рафсона. Метод Ньютона-Рафсона применяется для решения нелинейного уравнения, задающего условие нахождения наиболее близкой к идеалу точки в пространстве R^N . Данное условие задается для целевой функции, причем используется метрика L_p .

Второе ключевое отличие алгоритма AGE-МОЕА-II от его предшественников заключается в том, что для вычисления расстояний между недоминируемыми решениями предложено использовать величину,

построенную как приближенный аналог геодезического расстояния на искривленных топологических пространствах.

Псевдокод алгоритма, взятый из оригинального доклада, представлен на рисунке 3.

Algorithm 2: AGE-MOEA-II: An improved front modeling-based MOEA

Input: M : Number of objectives
 N : Population size
Result: Final population P

```
1 begin
2    $P \leftarrow \text{RANDOM-POPULATION}(N)$ 
3   while not (stop_condition) do
4      $Q \leftarrow \text{GENERATE-OFFSPRING}(P)$ 
5      $F \leftarrow \text{FAST-NONDOMINATED-SORT}(P \cup Q)$ 
6      $F \leftarrow \text{NORMALIZE}(F)$ 
7      $p \leftarrow \text{NEWTON-RAPHSON}(F_1)$  /* Eq. 9 */
8      $d \leftarrow 1$  /* First non-dominated rank */
9     while  $|P| + |F_d| \leq N$  do
10       $F_{\perp} \leftarrow \text{MANIFOLD-PROJECTION}(F_d, p)$ 
11       $D \leftarrow \text{GEODESIC-DIV}(F_{\perp}, p)$  /* Eq. 15 */
12      SURVIVAL-SCORE( $D, F, d, p$ )
13       $P \leftarrow P \cup F_d$ 
14       $d \leftarrow d + 1$ 
15    SORT( $F_d$ ) /* by survival scores */
16     $P \leftarrow P \cup F_d[1 : (N - |P|)]$ 
17  return  $P$ 
```

Рис. 3 – Псевдокод оригинального алгоритма AGE-MOEA2

Заметим, что в отличие от рисунка 1, на котором приведена процедура формирования одного поколения, код с рисунка 3 написан более крупными шагами и включает этап формирования начальной популяции. Реализация шагов описана в работе [16].

Вычислительные эксперименты

Для оценки результатов работы алгоритмов многокритериальной оптимизации среди разработчиков популярностью пользуется ресурс DTLZ [20, 21]. DTLZ – набор тестов для многокритериальных задач с масштабируемыми размерами приспособленности, получивший свое

название в честь авторов Деба, Тиле, Лауманна и Зицлера (Deb, Thiele, Laumanns and Zitzler). Все задачи в этом наборе тестов представляют собой непрерывные N -мерные многокритериальные задачи. Для каждой задачи определено количество локальных минимумов, а также известен глобальный минимум, что позволяет однозначно оценить результат работы алгоритма.

В данном исследовании тестирование проводилось на задаче DTLZ1. В DTLZ1 требуется найти решение задачи:

$$\begin{aligned} f_1(x) &= \frac{1}{2} x_1 x_2 \cdots x_{M-1} (1 + g(\bar{x}_M)) \rightarrow \min, \\ f_2(x) &= \frac{1}{2} x_1 x_2 \cdots (1 - x_{M-1}) (1 + g(\bar{x}_M)) \rightarrow \min, \\ &\dots \\ f_{M-1}(x) &= \frac{1}{2} x_1 (1 - x_2) (1 + g(\bar{x}_M)) \rightarrow \min, \\ f_M(x) &= \frac{1}{2} (1 - x_1) (1 + g(\bar{x}_M)) \rightarrow \min \end{aligned}$$

на множестве $\Omega = \{x \in R^N : 0 \leq x_i \leq 1, i = 1, 2, \dots, N\}$. Здесь через \bar{x}_M обозначен вектор, состоящий из $K = (N - M + 1)$ компонент, $\bar{x}_M = (x_M, x_{M+1}, \dots, x_N)$, а функция $g(\bar{x}_M)$ задана следующим образом:

$$g(\bar{x}_M) = 100 \left(|\bar{x}_M| + \sum_{i=M}^{N-M+1} ((x_i - 0,5)^2 - \cos(20\pi(x_i - 0,5))) \right).$$

Оптимальное по Парето решение данной задачи равно $x^* = (0,5; 0,5; \dots; 0,5)$, а значения целевой функции $(f_1(x^*), f_2(x^*), \dots, f_M(x^*))$ лежат на гиперплоскости

$$\sum_{i=1}^M f_i^* = 0,5.$$

Количество локальных минимумов целевой функции равно $(11N - 1)$, где N – длина цепочки ДНК особей генетического алгоритма, в связи с этим возникает сложность в сходимости к глобальному решению (оптимальному

Парето-фронт).

Для проведения вычислений авторами данной работы генетические алгоритмы NSGA-III и AGE-MOEA-II реализованы на языке программирования Rust. Язык Rust относительно новый язык программирования, его производительность сравнима с производительностью языка C++. Построение графиков выполнено с помощью библиотеки `plotters`. Для расчета опорных точек использован алгоритм Das-Dennis, так же реализованный в данной библиотеке.

В вычислительных экспериментах задавались следующие значения параметров задачи:

количество целей (размерность целевой функции) $M=3$;

длина цепочки ДНК $N=4$;

размер популяции – 91;

условие остановки – достигнута тысячная итерация.

Решения, найденные с помощью алгоритма NSGA-III, представлены на рисунке 4. На данном рисунке четко прослеживается упорядоченность найденных решений, область значений покрыта равномерно, глобальный минимум достигнут.

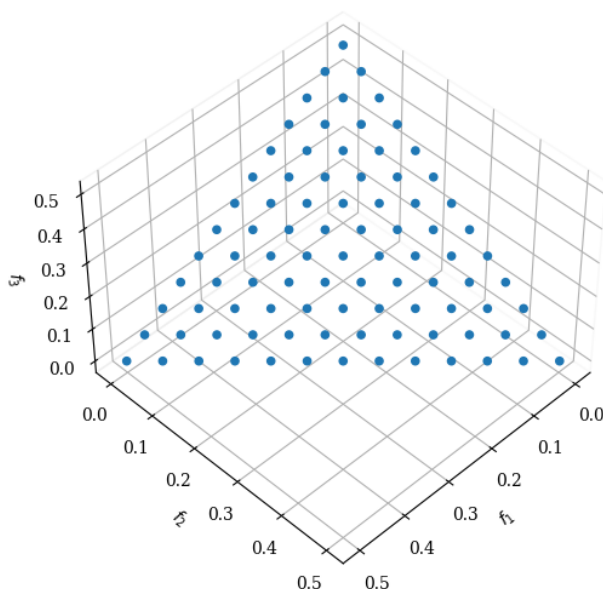


Рис. 4. – Парето-фронт, найденный с помощью NSGA-III

На рисунке 5 построены решения, полученные с помощью алгоритма AGE-MOEA-II.

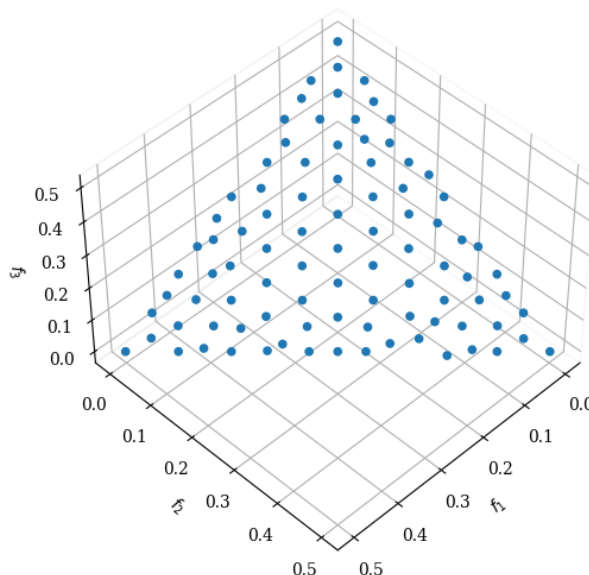


Рис. 5. – Парето-фронт, найденный с помощью AGE-MOEA-II

На данном рисунке также отчетливо видно, что глобальный минимум достигнут, однако решения упорядочены хуже, явно прослеживается неравномерное распределение по области значений.

Таким образом, можно сделать вывод, что в проведенных экспериментах NSGA-III показал себя более стабильным алгоритмом для решения задач с большим количеством локальных минимумов. Однако стоит обратить внимание, что для получения результатов NSGA-III использует опорные точки, а при увеличении размерности задачи количество опорных точек будет возрастать экспоненциально, что потребует огромного запаса оперативной памяти для выполнения программы. AGE-MOEA-II не зависит от количества целей, что позволяет решать более сложные задачи с меньшими вычислительными ресурсами. С другой стороны, если область допустимых значений известна, либо может быть предсказана с допустимой точностью, предпочтительнее использовать NSGA-III, так как AGE-MOEA-II

производит равномерное распределение решений по всей области лучшего Парето-фронта.

Заключение

В рамках проведенного исследования выполнен сравнительный анализ алгоритмов NSGA-III и AGE-MOEA-II, предназначенных для решения задач многокритериальной оптимизации. Результаты тестирования показали, что оба алгоритма эффективно справляются с задачами многокритериальной оптимизации, каждый демонстрируя свои уникальные преимущества в различных аспектах работы. Однако, поскольку анализ был ограничен только одним тестовым набором, выводы о полных возможностях алгоритмов остаются частичными. Это указывает на необходимость проведения более глубокого и разностороннего исследования с использованием различных тестовых наборов, что позволит более точно оценить их применимость и эффективность в различных условиях.

Полученные результаты важны как для теоретических исследований в области генетических алгоритмов, так и для практического применения в инженерии и других сферах, где многокритериальная оптимизация играет ключевую роль. Они способствуют дальнейшему развитию и улучшению методов многокритериальной оптимизации, открывая путь для новых исследований и новаций в этом направлении.

Литература

1. Дурницын О. А. Улучшение маршрутизации коммерческих автобетоносмесителей с использованием генетического алгоритма с акцентом на экономические и экологические показатели // Инженерный вестник Дона, 2021, № 12. URL: ivdon.ru/ru/magazine/archive/n12y2021/7327.

2. Азарнова Т. В., Гуров С. А., Недикова Т. Н. Генетический алгоритм формирования оптимального комплекса стратегий работы с договорами



портфеля просроченной задолженности // Инженерный вестник Дона, 2022, № 6. URL: ivdon.ru/ru/magazine/archive/n6y2022/7761.

3. Орловский Н. М. Решение задачи однокритериальной оптимизации процесса планирования действий экипажа Российского сегмента Международной космической станции на основе генетического алгоритма // Инженерный вестник Дона, 2013, № 3. URL: ivdon.ru/ru/magazine/archive/n3y2013/1776.

4. Микулик И. И., Благовещенская А. А. Распараллеливание гибридного алгоритма муравьиной колонии с изменяющимися с помощью генетического алгоритма параметрами // Проблемы информатики, 2023, № 2. С. 86-94. DOI: 10.24412/2073-0667-2023-2-86-97.

5. Матюхина Я. С. Нечеткая система управления параметрами генетического алгоритма, настраиваемая генетическим алгоритмом // Решетневские чтения, 2017, Т. 2. С. 219-220.

6. Кобак В. Г., Жуковский А. Г., Кузин А. П. Исследование применения одноточечного кроссовера при решении неоднородной минимаксной задачи // Инженерный вестник Дона, 2018, № 1. URL: ivdon.ru/ru/magazine/archive/n1y2018/4714.

7. Лотов А. В., Рябиков А. И. Простая эффективная гибридизация классической глобальной оптимизации и генетических алгоритмов многокритериальной оптимизации // Журнал вычислительной математики и математической физики, 2019, Т. 59, № 10. С. 1666-1680. DOI: 10.1134/S0044466919100107.

8. Акопов, А. С., Бекларян А. Л. Оптимизация стратегий поведения в имитационной модели многоагентной социально-экономической системы // Экономика и математические методы, 2023, Т. 59, № 3. С. 117-131. DOI: 10.31857/S042473880027006-5.

9. Пименов В. И., Пименов И. В. Анализ и визуализация данных в задачах

многокритериальной оптимизации проектных решений // Информатика и автоматизация, 2022, Т. 21, № 3. С. 543-571. DOI: 10.15622/ia.21.3.4.

10. Ногин В. Д. Принятие решений в многокритериальной среде: количественный подход. М.: Физматлит, 2005. 176 с.

11. Подиновский В. В., Ногин В. Д. Парето-оптимальные решения многокритериальных задач. М.: Физматлит, 2007. 256 с.

12. Тарутин А. В., Набатов А. В. Применение методов генетических алгоритмов для построения множества Парето в задачах многокритериальной оптимизации // Инженерный вестник Дона, 2015, № 4. URL: ivdon.ru/ru/magazine/archive/n4y2015/3359.

13. Deb K., Jain H. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints // IEEE Transactions on Evolutionary Computation 2014, 18, 4. Pp. 577–601. DOI: 10.1109/TEVC.2013.2281535

14. Hu C., Dai L., Yan X. [et al.] Modified NSGA-III for sensor placement in water distribution system // Information Sciences, 2020, Vol. 509. Pp. 488-500. DOI: 10.1016/j.ins.2018.06.055.

15. Jain Sh., Dubey K. K., Jain M., Rampal D. Evaluating NSGA-III : A Comprehensive Study on Multi-objective Optimization // International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2023. Pp. 146-150. DOI: 10.32628/cseit2390523.

16. Panichella A. An Improved Pareto Front Modeling Algorithm for Large-scale Many-Objective Optimization (1.0). Genetic and Evolutionary Computation Conference, Boston, 2022. DOI: 10.1145/3512290.3528732

17. Zhao B., Shen X., Peng K. [et al.] Adaptive computation offloading for latency-sensitive tasks in heterogeneous edge-cloud-enabled smart warehouses using Gau-Angle FIS and AGE-MOEA-II // Wireless Networks, 2023. DOI: 10.1007/s11276-023-03456-9.

18. Blank J., Deb K., Roy P. C. Investigating the normalization procedure of NSGA-III // In International Conference on Evolutionary Multi-Criterion Optimization, 2019. Pp. 229-240.
19. Panichella A. An adaptive evolutionary algorithm based on noneuclidean geometry for many-objective optimization // In Proceedings of the Genetic and Evolutionary Computation Conference, 2019. Pp. 595-603.
20. Deb K., Thiele L., Laumanns M., Zitzler E. Scalable test problems for evolutionary multi-objective optimization // TIK-Technical Report No. 112. 17.07.2001. 28 p. URL: research-collection.ethz.ch/bitstream/handle/20.500.11850/145762/eth-24696-01.pdf.
21. ParMOO API. The DTLZ Problem Library. URL: [p. readthedocs.io/en/stable/modules/dtlz.html](http://parmoo.readthedocs.io/en/stable/modules/dtlz.html).

References

1. Durnitsyn O. A. Inzhenernyj vestnik Dona, 2021, № 12. URL: ivdon.ru/ru/magazine/archive/n12y2021/7327.
 2. Azarnova T. V., Gurov S. A., Nedikova T. N. Inzhenernyj vestnik Dona, 2022, № 6. URL: ivdon.ru/ru/magazine/archive/n6y2022/7761.
 3. Orlovskiy N. M. Inzhenernyj vestnik Dona, 2013, № 3. URL: ivdon.ru/ru/magazine/archive/n3y2013/1776.
 4. Mikulik I. I., Blagoveshchenskaya A. A. Problemy informatiki, 2023, № 2. pp. 86-94. DOI: 10.24412/2073-0667-2023-2-86-97.
 5. Matyukhina YA. S. Reshetnevskiye chteniya, 2017, T. 2. pp. 219-220.
 6. Kobak V. G., Zhukovskiy A. G., Kuzin A. P. Inzhenernyj vestnik Dona, 2018, № 1. URL: ivdon.ru/ru/magazine/archive/n1y2018/4714.
 7. Lotov A. V., Ryabikov A. I. Zhurnal vychislitel'noy matematiki i matematicheskoy fiziki, 2019, T. 59, № 10. pp. 1666-1680. DOI: 10.1134/S0044466919100107.
-



8. Akopov, A. S., Beklaryan A. L. *Ekonomika i matematicheskiye metody*, 2023, T. 59, № 3. pp. 117-131. DOI: 10.31857/S042473880027006-5.
9. Pimenov V. I., Pimenov I. V. *Informatika i avtomatizatsiya*, 2022, T. 21, № 3. pp. 543-571. DOI: 10.15622/ia.21.3.4.
10. Nogin V. D. M. *Fizmatlit*, 2005. 176 p.
11. Podinovskiy V. V., Nogin V. D. M. *Fizmatlit*, 2007. 256 p.
12. Tarutin A. V., Nabatov A. V. *Inzhenernyj vestnik Dona*, 2015, № 4. URL: ivdon.ru/ru/magazine/archive/n4y2015/3359.
13. Deb K., Jain H. *IEEE Transactions on Evolutionary Computation* 2014, 18, 4. Pp. 577–601. DOI: 10.1109/TEVC.2013.2281535
14. Hu C., Dai L., Yan X. [et al.] *Information Sciences*, 2020, Vol. 509. Pp. 488-500. DOI: 10.1016/j.ins.2018.06.055.
15. Jain Sh., Dubey K. K., Jain M., Rampal D. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 2023. Pp. 146-150. DOI: 10.32628/cseit2390523.
16. Panichella A. *Genetic and Evolutionary Computation Conference, Boston*, 2022. DOI: 10.1145/3512290.3528732
17. Zhao B., Shen X., Peng K. [et al.] *Wireless Networks*, 2023. DOI: 10.1007/s11276-023-03456-9.
18. Blank J., Deb K., Roy P. C. In *International Conference on Evolutionary Multi-Criterion Optimization*, 2019. Pp. 229-240.
19. Panichella A. In *Proceedings of the Genetic and Evolutionary Computation Conference*, 2019. Pp. 595-603.
20. Deb K., Thiele L., Laumanns M., Zitzler E. *TIK-Technical Report No. 112*. 17.07.2001. 28 p. URL: research-collection.ethz.ch/bitstream/handle/20.500.11850/145762/eth-24696-01.pdf.
21. ParMOO API. URL: parmoo.readthedocs.io/en/stable/modules/dtlz.html.

Дата поступления: 10.04.2024

Дата публикации: 27.05.2024
