A Computationally Efficient Approach for Solving Lexicographic Multicriteria Optimization Problems

This paper proposes a computationally efficient approach for solving complex lexicographic multicriteria optimization problems in which efficiency criteria can be multiextremal and computing the values of criteria can be time-consuming. It is also assumed that in the course of calculations, how problems are formulated may change and consequently, solving dynamically defined sets of multicriteria optimization problems may become necessary. The proposed approach is based on reducing multidimensional problems to one-dimensional global optimization problems, utilizing efficient global search algorithms, and reusing the search information obtained in the process of calculations. Results of computational experiments demonstrate that the proposed approach can significantly reduce the computational complexity of solving multicriteria optimization problems.

*Keywords*: multicriteria optimization, lexicographic ordering, global optimization, dimensional reduction, search information, computational complexity

**1. Introduction**

Multicriteria optimization (MCO) problems are among the most common statements used to address decision-making problems and are widely found in applications. Multicriteria optimization is an area of vigorous scientific research, and during this time a large number of efficient methods of multicriteria optimization have been proposed and many practical problems have been solved (see, e.g., monographs [1-5]) and reviews of scientific and practical results in this field [6-9].

A key feature of MCO problems is the absence, in most cases, of a single decision which would best fulfill all efficiency criteria, due to possible inconsistencies. As a result, solving MCO problems usually requires finding several compromise (efficient and non-dominated) decisions that cannot be improved without damaging the effectiveness of some efficiency criteria in the MCO problem.

Нахождение всего множества эффективных решений может потребовать большого объема вычислений и часто бывает избыточным для decision maker. Как результат, в практике более часто используются подходы, позволяющие получить только некоторые ограниченные множества эффективных решений. Among these approaches is the scalarization approach, which applies methods involving the convolution of criteria to a single scalar criterion (see, e.g. [2, 14]). Iterative methods represent another approach [6, 10], when the decision maker is actively involved in the process of selecting decisions. An actively appropriated approach consists in developing and applying evolutionary algorithms based on the simulation of certain natural phenomena to solve MCO problems [10-13]. Перспективным направлением является также использование новой аксиоматики of [numerical infinities](http://wwwinfo.deis.unical.it/~yaro/Medals.pdf) для разработки методов решения задач MCO.

Among the widely used endeavors in solving MCO problems is lexicographic optimization methods can be singled out, when some ordering of criteria by importance is implemented and optimization is carried out based on their ordering [3]. При упорядоченности критериев по важности решение задачи MCO состоит в последовательной оптимизации критериев в порядке их важности: сначала оптимизируется первый, наиболее важный, критерий, затем на множестве решений, на которых первый критерий достигает наилучшие значение, проводится оптимизация второго критерия и т.д. Такая схема является простой и понятной, допускающей возможность применения многих эффективных алгоритмов скалярной оптимизации. Подобная многоэтапная схема может быть сведена к решению одной задачи скалярной оптимизации с помощью линейной свертки критериев [38]. Вместе с этим, такой подход чрезмерно уменьшает множество вычисляемых эффективных решений, что ограничивает возможность выбора для decision maker. Для того, чтобы расширить множество эффективных решений, которые могут быть рассмотрены decision maker, может быть использован ε-constraint scalarization подход, когда для оптимизации используется только самый важный критерий, а для остальных критериев задается максимально-допустимое значение (т.е. все остальные критерии переводятся в ограничения задачи MCO) [39,40]. В [41] расширенный вариант ε-constraint метода, в котором допускается нарушение ограничений, если это приводит к заметному улучшению значений по каким-либо критериям задачи MCO. Определенная сложность использования данного подхода состоит в том, что максимально-допустимые значения (допуска) критериев необходимо задавать изначально a priori до начала вычислений, что часто вызывает большие затруднения. Для устранения данной проблемы в [42] предлагается сочетать исходную последовательную схему решения задач лексикографической оптимизации, когда допуска можно будет задавать в процессе вычислений после завершения очередного этапа оптимизации (правда такая схема снова приводит к необходимости решения последовательности задач скалярной оптимизации). Еще большая свобода по формулированию задач поиска эффективных решений дается в рамках подхода [43], когда порядок важности критериев также может меняться динамически. Следует отметить, что возможность вариации постановки задачи поиска эффективных решений носит принципиальный характер, поскольку требования к оптимальности выбираемых решений могут меняться у decision maker в процессе вычислений. Возможность подобной вариации обеспечивается и при использовании других походов решения задачи MCO на основе различных методов скаляризации критериев – см., например, [31,44]. Комбинированный вариант использования скаляризации критериев и дополнительных ограничений ε-constraint метода предложен в [1].

This work is devoted to solving lexicographic multicriteria optimization (MCOlex) problems that arise when designing complex technical objects and systems. In such applications, efficiency criteria may have a complex multiextremal form, and calculating criteria values may require large amounts of computations. In such conditions, finding even one efficient decision requires significant calculations, while determining several (or a whole set) of efficient decisions becomes an issue of great computational complexity.

The properties listed above highlight a key feature of the MCOlex problems ‑ high computational complexity. One promising attempt to solve such problems is using the model-based approach, when, after a small number of calculations determining the values of time-consuming criteria and, fast-computable approximating functions are constructed [15-16]. This approach is quite efficient; however, it is difficult to construct good approximations given the multiextremal behavior of efficiency criteria.

This paper proposes a solution to the time-consuming class of MCOlex problems by using an approach based on the following main points. First of all, the problem is reduced to solving a sequence of global optimization problems with nonlinear constraints (GCO) [2,14]. Then, efficient global search algorithms developed in the framework of the information-statistical theory of multiextremal optimization are applied for solving the GCO problems [17-18]. And finally, the search information obtained in the process of solving the MCOlex problem is fully utilized. In general, the developed approach allows us to significantly reduce the amount of executed computations, up to just a few iterations when searching for new efficient decisions.

The further structure of the paper is as follows. Section 2 presents a new class of optimization problems ‑ multistage lexicographic multicriteria optimization (MMCOlex) problems ‑ whose solution is reduced to solving a sequence of global optimization problems with nonlinear constraints. Section 3 considers the basics of the approach that has been developed: reducing multidimensional MCOlex problems to one-dimensional global optimization problems, applying efficient global search algorithms developed in the framework of the information-statistical theory of multiextremal optimization, and reusing the search information obtained in the process of calculations. Section 4 provides a theoretical analysis of the effectiveness of reusing the search information for solving MMCOlex problems. Section 5 contains the results of numerical experiments. The paper concludes by discussing the results that were obtained and suggests some main directions for further research.

**2. Problem statement**

The multicriteria (or vector) optimization (MCO) problem can be defined as follows:

|  |  |
| --- | --- |
| *f*(*y*) = (*f*1(*y*), *f*2(*y*),…, *f*s(*y*)) → min, *y*∈*D*, | (1) |

where *f*(*y*) = (*f*1(*y*), *f*2(*y*),…, *f*s(*y*)) is the vector efficiency criterion, *y* = (*y*1, *y*2,…, *yN*) is the vector of variable parameters, and *N* is the dimension of the multicriteria optimization problem to be solved. The search domain *D* defines the set of possible parameter values and is usually an *N*-dimensional hypercube

|  |  |
| --- | --- |
| *D*  = { *y*∈*RN*: *ai*≤ *yi*≤ *bi*, 1≤*i*≤*N* } | (2) |

for the given boundary vectors *a* and *b*.

Without a loss of generality, it is assumed that the values of the efficiency criteria are not negative and their decrease corresponds to an increase in the effectiveness of the considered decisions *y*∈*D*. Problem (1) is considered in relation to the most difficult decision-making problems in which the efficiency criteria *fi*(*y*), 1≤*i*≤*s,* can be significantly multiextremal, and the procedures for calculating the values of efficiency criteria at points of the search domain *y*∈*D* can be computationally expensive. It is also assumed that the criteria *fi*(*y*), 1≤*i*≤*s*, satisfy the Lipschitz condition

|  |  |
| --- | --- |
| , 1≤*i*≤*s,* | (3) |

where *Li* is the Lipschitz constants for the function *fi*(*y*), 1≤*i*≤*s* and  denotes the Euclidean norm in . Fulfillment of the Lipschitz condition means that for small variations of the decision *y*∈*D*, corresponding changes in the values of the criteria *fi*(*y*), 1≤*i*≤*s,* are limited.

|  |  |  |
| --- | --- | --- |
| In problem (1), the efficiency criteria are usually contradictory and there is no decision *y*∈*D*, that would provide the best (smallest) values for all criteria at the same time, i.e.   |  |  | | --- | --- | | *.* | (4) | |

If the relation (4) is valid when solving the MCO problem, then decisions ∈*D* can be determined, at which the values of the criteria cannot be improved without deterioration of the values of some criteria *fi*(*y*), 1≤*i*≤*s*, that is

|  |  |
| --- | --- |
| *y*'∈*D* : (*fi*(*y*')≤*fi* (), 1≤*i*≤*s*), ( : *fj*(*y*')<*fj* ()). | (5) |

Such non-improvable decisions are called Pareto efficient or optimal.

The Pareto set *P*(*f*,*D*) of all efficient decisions can be quite large, which makes it difficult for the decision maker to analyze. A possible way to reduce the number of efficient decisions under consideration is to assume that the effectiveness criteria are ordered in terms of importance, which is often the case in practical applications.

Without a loss of generality, we will further assume that the criteria *fi*(*y*), 1≤*i*≤*s* are ordered in importance according to their enumeration in (1). The ordering of the criteria determines the relative linear order in the search domain *D*, that is

|  |  |
| --- | --- |
| . | (6) |

As a result, problem (1) is reduced to the lexicographic multicriteria optimization problem (MCOlex)

|  |  |
| --- | --- |
| *f*(*y*) = (*f*1(*y*), *f*2(*y*),…, *f*s(*y*)) → minlex, *y*∈*D*, | (7) |

the solution of which is carried out in stages: initially, the first (most important) criterion is optimized; next, if the solution is not the only one, then the second criterion is optimized on the set of solutions of the first stage, and so on.

|  |  |
| --- | --- |
| , | (8) |

where *Arg* means the set of all decisions that achieve the minimum value of the optimized criterion. It should be noted that the sequence of steps in (8) may not be fully completed if at some stage *l*, , the set degenerates and contains only a single decision *у*∈*D*.

The set of efficient decisions *Plex*(*f*,*D*) obtained as a result of (6) is a subset of the Pareto domain *P*(*f*,*D*) and can contain from one to several decisions ∈*D* in the case that the minimum value of the last criterion is achieved at several points in the search domain *D*. Such a sharp reduction in the set of the efficient decisions being considered may be undesirable. Expanding the set of *Plex*(*f*,*D*) decisions can be achieved through somewhat weakening the “strict” lexicographic order and by allowing, at each stage of the computational scheme (8), the choice of decisions in a certain neighborhood of the minimum values of the optimized criterion, that is

|  |  |
| --- | --- |
|  | (9) |

This approach is also widely known as the ε-constraint method (известный также как the method of successive concessions) [39-42]. The choice of feasible concessions *δi*, 1≤*i*≤*s* in the computational scheme (9) allows one to obtain any efficient decision ∈ *Plex*(*f*,*D*) and take into account the peculiarities of the MCOlex problem being solved. At the same time, this approach makes it necessary, at each stage of scheme (9), to solve more complex global optimization problems with nonlinear constraints.

It should also be noted that during the calculation process, it may be necessary to change the selected values of the feasible concessions *δi*, 1≤*i*≤*s*; the concessions may be quite rigid or, conversely, excessively large. In general, one may need to change the order of importance in efficiency criteria. Such assumptions necessitate a more general formulation of how to solve the MCOlex problem and to provide possible solutions to many problems of the form (9)

|  |  |
| --- | --- |
|  | (10) |

which can be changed dynamically during calculations by adding new or deleting existing multicriteria optimization problems. In general, such an approach defines a new class of optimization problems: multistage lexicographic multicriteria optimization problems (MMCOlex).

# **3. The Proposed Approach: Key Ideas and Methods**

The extended formulation of lexicographic multicriteria optimization problems (7)-(8) involves multiple solutions of global (GO) optimization problems with nonlinear constraints. Problems of this type are computationally time-consuming and subject to the “curse of dimensionality” ‑ computational complexity increases exponentially with increasing dimensionality.

Область глобальной оптимизации привлекает внимание ученых на протяжении многих последних лет, результаты исследований опубликованы в большом количестве монографий – см., например, [17-25]. Обоснованный выбор подхода для решения задач GO среди всего большого множества уже существующих алгоритмов глобального поиска может быть выполнен с учетом следующих требований:

* Поскольку оценка глобального минимума является интегральной характеристикой решаемой задачи оптимизации (значение минимизируемой функции в точке глобального минимума должно быть сопоставлено со значениями функции во всех других точках области поиска), методы глобального поиска должны обеспечивать какое-либо покрытие области поиска,
* В силу «проклятия размерности» для снижения вычислительной сложности получаемое покрытие области поиска должно быть неоднородным (nonunoform), т.е. должно быть плотным только в окрестности глобально оптимальных точек,
* Возможность построения неоднородных покрытий может быть обеспечена только при наличии каких-либо предположений a priori о поведении минимизируемой функции – одна из наиболее часто используемых форм таких предположений состоит в выполнимости условия Лищица (2),
* Построение неоднородных покрытий области поиска предполагает анализ многомерной поисковой информации (точки выполненных итераций глобального поиска и значений минимизируемой функции в этих точках) очень большого объема, что значительно усложняет сложность и вычислительную трудоемкость методов глобальной оптимизации – для снижений подобной вычислительной сложности в глобальной оптимизации достаточно часто используется редукция размерности на основе, например, диагонального расширения одномерных алгоритмов [22, 45], the nested optimization scheme [46,47], dimension reduction using Peano space-filling curves [17,23],
* При многократном решении задач multistage lexicographic multicriteria optimization методы глобального поиска могут уменьшить сложность вычислений за счет использования поисковой информации, получаемой в процессе вычислений.

С учетом перечисленных выше требований the proposed approach for solving MMCOlex problems is based on three key ideas:

* Reduction of multidimensional multiextremal optimization to problems of one-dimensional global search by using Peano space-filling curves;
* Using the global optimization methods, that generate nonuniform coverage of search domain;
* Intensive reusing search information obtained in the process of computations, for solving multiple global optimization problems.

**3.1. Dimension reduction of multidimensional multicriteria optimization problems**

The nonuniform coverage of the search domain D can be constructed adaptively in the following manner: when choosing points for the next iterations, the global search is performed taking into account all the available search information (points of previous search iterations and values of the optimized function at these points), that is

|  |  |
| --- | --- |
| , | (11) |

where *k* is the number of the global search iterations being performed, is the decisive rule of the global search algorithm, according to which the point of the next iteration is selected, and is the search information available at the *k* iteration of the global search. The computational complexity of the decisive rule can be significantly simplified by reducing the optimization problems to be solved using Peano space-filling curves or evolvents *y*(*x*), that uniquely and continuously map the interval [0,1] to an *N*-dimensional hypercube *D* (see, e.g., [17-18, 23]). As a result of this reduction, the initial multidimensional multicriteria optimization problem (7) is reduced to a one-dimensional problem:

|  |  |
| --- | --- |
| *f*(*y*(*x*)) = (*f*1(*y*(*x*)), *f*2(*y*(*x*)),…, *f*s(*y*(*x*))) → minlex, x∈[0,1].. | (12) |

It should be noted that the one-dimensional functions obtained as a result of the reduction satisfy the uniform Hölder condition (see [17–18]), that is

|  |  |
| --- | --- |
| , | (13) |

where the constants *Hi* are determined by the ratio , 1≤*i*≤*m,* *Li* are the Lipschitz constants from (2), and *N* is the dimension of the optimization problem (1).

Using dimension reduction the search information can be converted from (9) to the form

|  |  |
| --- | --- |
| , | (14) |

in which, in addition to the reduced representation, the data is arranged in ascending order[[1]](#footnote-1) of points, that is

|  |  |
| --- | --- |
|  | (15) |

for a more efficient execution of global search algorithms.

Dimension reduction using Peano curves is widely used in the development of efficient global search algorithms in the framework of the information-statistical theory of multiextremal optimization [17,18,23,26-27]. Numerical methods for constructing approximations of Peano curves (evolvents) with a given accuracy are considered in [17-18].

3**.2. An efficient method for solving global optimization problems with nonlinear constraints**

The problems solved in the framework of the computational scheme (9) are one-dimensional global optimization problems with nonlinear constraints

|  |  |
| --- | --- |
| min {ϕ(x) : gi(x)≤0, 1≤*i*≤*m*,x∈=[0,1] } | (16) |

(the notation gm+1(x) = ϕ(x) will also be used later).

The presence of nonlinear constraints significantly complicates the complexity of solving global optimization problems; the resulting solutions must belont to a feasible domain

|  |  |
| --- | --- |
| . | (17) |

To reduce the complexity of solving optimization problems with constraints, simpler cases are often highlighted; for example, problems with linear or quadratic constraints are selected. Various methods of approximating complex constraints with the aid of simpler forms (linear, convex, etc.) are widely used. However, the penalty function method is most commonly used, when calculating in an unfeasble region Δ\, some “penalty” are added to the values of the minimized function. But the penalty function method often involves repeatedly solving the optimization problem to determine a sufficient value of the penalty coefficient. More complete information on methods for solving optimization problems with constraints can be obtained, for example, in monographs [28–29].

The proposed approach is based on the original method of separate accounting for constraints developed in the framework of the information-statistical theory of multiextremal optimization [18]. The key idea of the approach consists in constructing a certain integral objective function without constraints, the solution of which also provides a solution to the initial constrained optimization problem (16); a more detailed presentation of this method is given below.

To provide uniform calculations a sequential scheme is applied to obtain values the constraints gi(y(x))≤0, 1≤*i*≤*m*, and the minimized function ϕ(x). According to this scheme the constraint values are calculated strictly in the order of their numbers, and the calculations immediately stop when the first violated constraint is detected (the partial computability scheme). The number of constraints in which the values were calculated will be referred below as the index ν=ν(x), that is

|  |  |
| --- | --- |
| ν(x) : gν(x)>0, gj(x)≤0, 1≤j≤ν−1, 1≤ν=ν(x)≤m | (18) |

(when all the constraints are satisfied, ν=m+1 is assumed). Introducing the index concept allows us to determine the integrated function for the problem (16) (see Fig. 1)

|  |  |
| --- | --- |
| F(x) = gν(x), ν=ν(x), | (17) |

defined and computable everywhere in [0,1]*.* Its value at the point x∈[0,1] is either the value of the left side of the constraint violated at this point (the case when ν≤ m), or the value of the minimized function (the case when ν=m+1). The process of calculating the value of F(x), x∈[0,1], is completed either as a result of establishing the inequality gj(x)>0, or as a result of reaching the value ν(x)=m+1 (below, this procedure is called a *trial*).

|  |  |
| --- | --- |
|  |  |

а) б)

Fig. 1. An integrated function for the problem (16) using a sequential constraint calculation scheme: a) the function *F*(*x*) from (19), b) the function from (20)

The function F(x) consists of separate fragments of the constraints gi(x)≤0, 1≤*i*≤*m*, and the minimized function ϕ(x). For the homogeneity of the components, then the function F(x)is converted to the form

|  |  |
| --- | --- |
|  | (20) |

where *M* is the constraint number for which there are no feasible points (the maximum value of the index ν(x)), is the value of the minimum constraint violation [[2]](#footnote-2), and ,1≤ν≤*M*, are the Hölder constants from (13). As a rule, these values are a priori unknown, but when performing calculations, instead of these values, their adaptive estimates can be applied. For this, the search information from (14) should be expanded with data

|  |  |
| --- | --- |
| , | (21) |

where Then the estimates of the required values can be determined as follows

|  |  |
| --- | --- |
| Mk = max {ν=ν(xi), 1≤i≤k }, | (22) |
|  | (23) |
| . | (24) |

If in (24) turns out to be indefinite or equal to zero, then =1is taken. For the initial values of these values, M0 = =1, =0, can be taken.

In the framework of the proposed approach to minimize the function the global search algorithm for multiextremal optimization problems with nonlinear constraints[[3]](#footnote-3) (AGCS) is applied. The general computational scheme of this method can be presented as follows [18].

The first trial is carried out at an arbitrary internal point x1∈(0,1)*.* The choice of the point xk+1, k≥1 of any next trial is determined by the following rules.

*Rule* 1. For each interval (xi−1,xi), 1<i≤k, in the set calculate the characteristic

|  |  |
| --- | --- |
| , | (25) |

where the expression is set by AGCS.

*Rule* 2. Determine the interval (xt−1,xt) that corresponds to the maximum characteristic

|  |  |
| --- | --- |
| R(t)=max{R(i): 1<i≤k}. | (26) |

*Rule* 3. Execute a new trial at the point *x*k+1∈(xt−1,xt), determined in accordance with the expression

|  |  |
| --- | --- |
| , | (27) |

where the expression is also set by AGCS.

Iterations of the algorithm are terminated if the stop condition is satisfied

|  |  |
| --- | --- |
| ρt≤ε, | (28) |

where *t* from (26), and ε>0is the given accuracy.

Results of using the AGCS algorithm to solve the test problem from Fig. 1 with the accuracy ε=10−5are shown in Fig. 2. The coordinates of the trial points performed by the algorithm in the process of solving the problem are marked in Fig. 2 by three rows of vertical strokes. The strokes in the upper row correspond to points with a unit index, the second row to points whose indices equal to 2; the points marked with strokes in the bottom row belong to the feasible domain. The coordinates of trials performed at close points are marked with a dark rectangle. In total, the value of the first constraint was computed 147 times, the value of the second constraint - 84 times, and the value of the minimized function was calculated only 35 times.

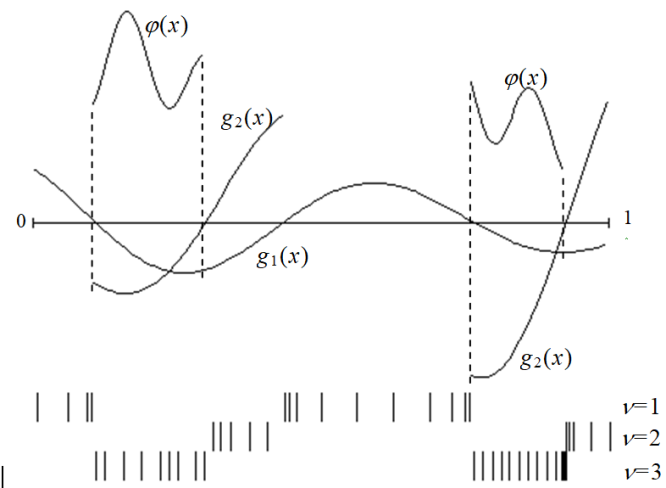


Fig. 2. Results of using the AGCS algorithm to solve the test problem

Various modifications of this algorithm and the corresponding theory of convergence are presented in [18]. For instance, the following theorem is valid.

**Theorem 1.** If the AGCS algorithm is used to solve problem (16), and the following conditions are satisfied:

1) the functions gj, 1≤j≤m+1,  [0,1] satisfy the Hölder condition of (13) with the constants Hj, 1≤j≤m+1,

2) for the values hν from (24), starting from a certain step, the inequalities are valid[[4]](#footnote-4)

|  |  |
| --- | --- |
| rνhν>2Hν, rν>1,1≤ν≤m+1, | (29) |

then the set of limit points of the sequence {xk}, generated by the algorithm coincides with the set of global minimum points of the problem (16) with ε=0 in the stop condition (28).

**3.3. Acceleration of computations based on the reuse of search information**

As noted earlier, the MCOlex problem at each stage (9) of the the ε-constraint method is a global optimization problem with nonlinear constraints (16). Generally, the solution to such problems must start at the beginning, and multiple solutions of such global optimization problems may require extensive calculations. However, the presence of search information from (21) allows us to bring the results of previous calculations to the values of any next solved optimization problem in the scheme (9) without any time-consuming calculations of the values of the criteria of the problem (1) [30-31]. This reuse of search information to find the next efficient decision reduces the amount of computations needed to solve each next optimization problem to just a few iterations - see Section 5 for the results of computational experiments.

Moreover, using search information allows to reduce the multistage solution of the MCO problem in accordance with the scheme (9) to the solution of the only global optimization problem in the last stage of the scheme (9), namely

|  |  |
| --- | --- |
|  | (30) |

where for the unknown a priori values can be used the estimates from (23) obtained from the search information, and for the concessions the normalized values can be applied, that is

|  |  |
| --- | --- |
| . | (31) |

In particular, a similar one-step scheme (30) - (31) can be applied through the use of AGCS

With a complete scheme for calculating criteria values, i.e. when at the iteration points , the values of all criteria are calculated, the effect of using the search information can be even more significant. In this case, the search information can also be used when changing the values of the concessions and thus, the solution of the next problem from the family (10) can be carried out repeatedly using the results of all previously performed calculations. The results of computational experiments (see Section 5), demonstrate that the volume of computations performed, based on reusing search information, can significantly decrease.

The AGCS algorithm, supplemented by the ability to use search information to solve multiple MCOlex problems, will be referred to below as the global search algorithm for multistage solutions of a set of multiextremal optimization problems with nonlinear constraints (MAGCS).

**4. Оценка эффективности многократного использования поисковой информации**

Оценка эффективности многократного использования поисковой информации основывается на ключевых свойствах используемого алгоритма AGCS, который при поиске глобального минимума конструирует сетку из точек поисковых испытаний, покрывающих области поиска. Кроме того, алгоритм является устойчивым, в результате чего при небольших вариациях минимизируемой функции величина изменения оценки глобального минимума является ограниченной.

Для снижения сложности проведения теоретического анализа все дальнейшие исследования выполнены для одномерного варианта алгоритма AGCS; полученные результаты в силу использования редукции размерности будут справедливыми и при решении многомерных задач оптмизации.

Начальный теоретический анализ свойств глобального поиска при использовании поисковой информации был выполнен в [48]. Полученные теоретические утверждения могут быть переформулированы применительно к задачам lexicographic multicriteria optimization problems.

**Теорема 2**. Пусть очередная решаемая задача оптимизации вычислительной схемы (9) отличается от предыдущей решенной задачи на некоторую ограниченную величину , т.е.

|  |  |
| --- | --- |
|  | (32) |

и пусть при минимизации функции было выполнено условие остановки (28) для заданной точности глобального поиска . Тогда

|  |  |
| --- | --- |
| , | (33) |

если начиная с некоторой итерации минимизации функции для оценки константы Липщица *m* и некоторого выполняется условие

|  |  |
| --- | --- |
| , | (34) |

где

|  |  |
| --- | --- |
| , | (35) |
| , из (3), | (36) |
| , | (37) |
| , | (38) |

(точки , в (38) берутся из поисковой информации из (21), полученной при минимизации функции , а есть величины из (22), (24) и (29) соответственно,).

Это утверждение означает, что если погрешность из (33) определения минимального значения очередной решаемой задачи оптимизации является приемлемым, то для минимизации функции не требуется проведения каких-либо дополнительных итераций глобального поиска - оценки минимального значения можно получить в соответствии с (33), используя значения , расположенные в поисковой информации из (21), полученной при минимизации функции .

**Теорема 3**. Пусть решение задачи оптимизации вычислительной схемы (9) проведено до выполнения условия остановки (28) для заданной точности глобального поиска и для интервала достигнута плотность испытаний

|  |  |
| --- | --- |
|  | (39) |

где есть количество точек испытаний, попавших в интервал при минимизации функции . Тогда при решении очередной задачи оптимизации с использованием поисковой информации из (21), полученной при минимизации функции , количество испытаний в том же самом интервале вместо максимальной оценки

|  |  |
| --- | --- |
| , *m* из (35), | (40) |

будет ограничено величиной

|  |  |
| --- | --- |
|  | (41) |

при выполнении условия

|  |  |
| --- | --- |
| . | (42) |

Учитывая оценку (41) максимального количества испытаний в подинтервалах области поиска можно сформулировать следующее утверждение.

**Теорема 4**. Если при решении задачи оптимизации вычислительной схемы (9) количество испытаний в интервале превышает максимальную оценку (41), то при решении очередной задачи оптимизации с использованием поисковой информации из (21), полученной при минимизации функции дополнительные испытания в интервале проводиться не будут при выполнении условия (42).

Сформулированные выше теоретические утверждения позволяют оценить положительный эффект от многократного использования поисковой информации, получаемой в процессе вычислений. Если новая решаемая задача является достаточно близкой к уже решенным задачам, то оценка глобального поиска может быть получена без выполнения дополнительных итераций глобального поиска на основе имеющейся поисковой информации (Теорема 2). В предельном случае, при решении достаточного большого множества информационно-связанных задач оптимизации, каждая новая решаемая задача может оказаться близкой к одной из уже ранее решенных задач, и для решения этой новой задачи достаточно уже имеющейся поисковой информации. В иных случаях, когда новая решаемая задача оптимизации достаточно сильно отличается от ранее уже решенных задач, использование поисковой информации, ранее полученной в ходе вычислений, позволяет сократить количество испытаний, выполняемых в подинтервалах области поиска - вплоть до полного отсутствия дополнительных испытаний в этих подинтеравалах (Теоремы 3-4).

**5. Results of computational experiments**

The computational experiments were carried out on the Lobachevsky supercomputer at Nizhny Novgorod State University (the operating system CentOS 6.4, the management system SLURM). A supercomputer node has 2 Intel Sandy Bridge E5-2660 2.2 GHz, 64 Gb RAM processors. Each CPU is 8-core; that is, there are 16 CPU cores available on the node. The Intel C ++ 14.0.2 compiler was used to obtain an executable program code. The numerical experiments were performed using the Globalizer system [32].

Оценка эффективности решения задач многокритериальной оптимизации является не совсем очевидной задачей. Если в случае скалярной (однокритериальной) оптимизации при оценке эффективности можно ограничиться показателями точности решения и количества выполненных оптимизационных итераций (количества вычислений значений минимизируемой функции), то в случае задач МКО результатом вычислений является уже несколько эффективных решений или даже аппроксимация всего множества Парето. Существуют множество различных подходов – так, например, в качестве показателей эффективности предлагается использовать mutual domination rate [49], degree of approximation [50], overall Pareto spread [51], etc. В [52] предлагается даже большой спектр из 57 показателей эффективности решения задач МКО. К сожалению, следует отметить, какого-то единого подхода для оценки эффективности методов МКО еще не принято.

В данной статье для оценки результатов вычислительных экспериментов используются два широко используемых показателя: the completeness and uniformity of the Pareto domain coverage were compared, using the following two indicators [35-36]:

* The hypervolume index (HV). This indicator characterizes the approximation of the Pareto domain in terms of completeness; a higher value corresponds to a more complete coverage of the Pareto domain.
* The distribution uniformity index (DU). This indicator characterizes the uniformity of Pareto domain coverage; a lower value corresponds to a more uniform coverage of the Pareto domain.

В последующих экспериментах в силу того, что решение задачи МКО сводится к решению множества задач скалярной оптимизации, для оценки результатов будут использоваться «классические» показатели: точность решения и количество вычислений значений минимизируемой функции. Последний показатель тем более важен, поскольку в силу исходного предположения о высокой трудоемкости вычисления значений критериев эффективности задачи МКО данный показатель характеризует вычислительную сложность решения задачи МКО.

Другой важный аспект при оценке эффективности решения задачи состоит в выборе тестовых задач, на примере которых будет проверяться эффективность используемых методов. К сожалению, здесь тоже пока нет единого подхода. Во многих исследованиях эффективность разработанных методов демонстрируется на примере крайне ограниченного набора тестовых задач. Для исправления такой ситуации предпринимаются многократные попытки формирования достаточно широких наборов тестовых задач, которые могут быть использованы при оценки эффективности методов решения задач МКО. Так, в [53] предлагается a flexible toolkit for constructing well-designed test problems. В [54] представлен набор из 24 тестовых функций, которые широко используются при сравнении методов МКО. В [55] предлагается набор из 100 тестовых задач. В [56] предлагается подход, при котором набор критериев эффективности задачи МКО формируется из известных скалярных тестовых функций и, тем самым, может генерироваться практически неограниченное множество тестовых задач МКО. Вместе с этим, набор тестовых задач МКО, в которых бы критерии эффективности были многоэкстремальными функциями, достаточно ограничен. Кроме того, предлагаемые наборы состоят, как правило, из единичных примеров задач МКО, что затрудняет построение достоверных оценок эффективности методов.

В данной статье применяется подход для оценки эффективности предлагаемых методов на основе массового[[5]](#footnote-5) решения задач МКО с использованием генераторов, позволяющих порождать неограниченно задачи МКО с использованием какого-либо случайного механизма. В качестве такого источника таких задач используется широко используемый генератор GKLS [37], который позволяет порождать многоэкстремальные функции с заранее определенными характеристиками (dimensionality, the number of local minima, the size of their attraction domains, the point of global minimum, and the value of the functions therein, etc.). На рис. 1 представлен пример двух критериев, сгенерированных с использованием GKLS.

a)

б)

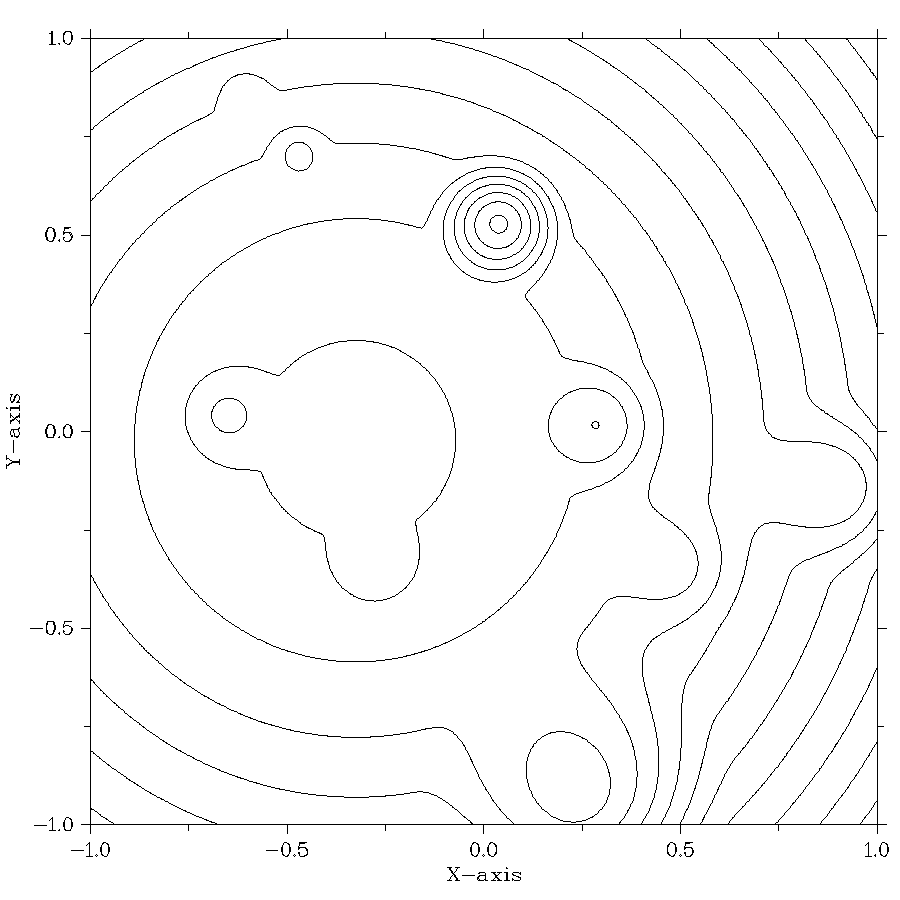
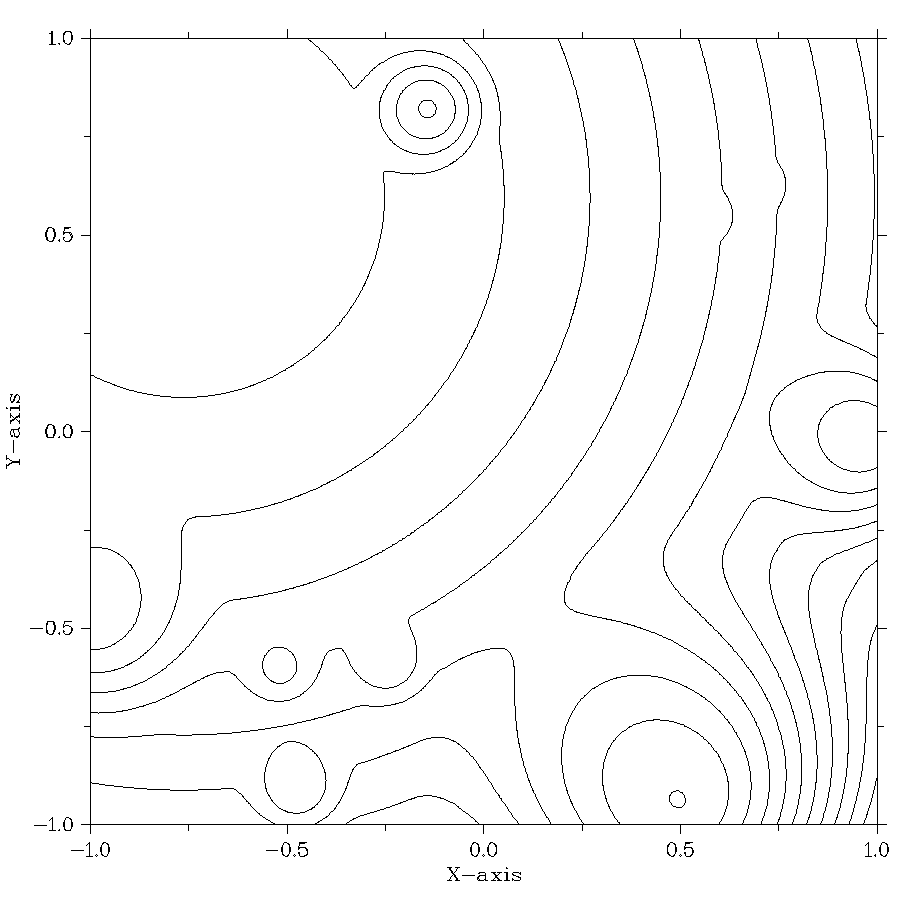


Рис.1. Линии уровня для двух критериев, полученных генератором GKLS

Как уже отмечалось ранее, the global optimization algorithms used within the framework of the proposed approach were developed in in the framework of the information-statistical theory of multiextremal optimization. These methods have proven to be efficient in the process of conducting computational experiments and have been widely used in solving practical global optimization problems (see, e. g., [17,18,23,26-27,33-34]). The results of computational experiments in solving multicriteria optimization problems are given below.

In the first series of computational experiments, the developed MAGCS algorithm was compared with several multicriteria optimization algorithms. To make a comparison, we used the test two-criteria problem proposed in [35]:

(43)

In the framework of this experiment, five multicriteria optimization algorithms were compared: the Monte-Carlo (MC) method, the genetic algorithm SEMO from the PISA library [9, 36], the Non-uniform coverage (NUC) method [35], the bi-objective Lipschitz optimization (BLO) method [36], and the MAGCS algorithm proposed in this paper. The results of solving problem (29) for all these methods (except for MAGCS) were obtained in [36].

For MAGCS, 50 problems (3) were solved for different values , uniformly distributed in the interval [0,1]. The accuracy ε=0.05 from (28) and the reliability parameters r1= r2=3.0 from (29) were used. The results of the executed experiments are presented in Table 1.

Table 1. The numerical results of solving the multicriteria optimization problems from (28)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | MC | SEMO | NUC | BLO | **MAGCS** |
| **Number of executed iterations** | 500 | 500 | 515 | 498 | **273** |
| **Number of points in Pareto domain approximation** | 67 | 104 | 29 | 68 | **80** |
| **HV index** | 0.300 | 0.312 | 0.306 | 0.308 | **0.314** |
| **DU index** | 1.277 | 1.116 | 0.210 | 0.175 | **0.096** |

The results from the executed experiments demonstrate that the MAGСS algorithm has a significant advantage in comparison with the considered multicriteria optimization methods of, even when solving relatively simple MCO problems.

In the second series of computational experiments, two-criteria two-dimensional MCO problems were solved. Multiextremal functions obtained using the GKLS generator [37] served as the criteria for the problem. During the experiments, 100 multicriteria problems of this class were solved. In each problem, Pareto-optimal decisions were searched for 50 different values of , uniformly distributed in the interval [0,1]; (that is, 5,000 global optimization problems were solved). The results obtained were averaged over the number of MCO problems solved.

In the computational experiments, calculations are terminated when the required accuracy was achieved. When the process was terminated, the solution was evaluated for correctness. As a means of control, we compared the solution points found by MAGCS and the points calculated from the Pareto boundary, taking into account the selected values of The method parameters were set as foolows: the accuracy ε = 0.02 and the reliability parameters Results of the computational experiments are presented in Table 2.

In Table 2, the first and fifth columns indicate the average number of iterations executed by the MAGCS algorithm for solving the MCOlex problems. The second and sixth columns contain the percentage of completed problems. The third, fourth, seventh and eighth columns show the values of HV and DU indicators. The last column gives values showing the reduction in the number of global search iterations that were executed when solving the MCOlex problems, by reusing search information.

Table 2. Results of the series of experiments to solve two-dimensional two-criteria MCO problems

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Search information** | | | | | | | | **Reduction in iteration number** |
| **not used** | | | | **used** | | | |
| **Method iterations** | **Problem  solved** | **Avg HV** | **Avg DU** | **Method  iterations** | **Problem solved** | **Avg HV** | **Avg DU** |
| 41 710.1 | 98.0% | 33.43 | 0.173 | 2 407.41 | 99.3% | 33.34 | 0.224 | 17.3 |

Results obtained from these experiments show that the reuse of search information can reduce the amount of calculations by 17.3 times without expending additional computational resources, and according to the average HV and DU indicators, the quality of the Pareto domain remains on average at the same level.

In the third series of computational experiments, the two-criteria four-dimensional MCO problems were solved. As in the second series of computational experiments, 100 multicriteria problems were solved. In each problem, the Pareto-optimal decisions were calculated for 50 different values of θj, 1≤j <s, uniformly distributed in the interval [0,1]. The criteria of the MCO problems being solved were determined using the GKLS generator [37]. The method parameters were set as follows: the accuracy ε = 0.025 and the reliability parameters The results of computational experiments are presented in Table 3.

Table 3. Results of the series of experiments to solve two-criteria four-dimensional MCO problems

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Search information** | | | | | | | | **Reduction in iteration number** |
| **not used** | | | | **used** | | | |
| **Method iterations** | **Problem**  **solved** | **Avg HV** | **Avg DU** | **Method iteration** | **Problem Solved** | **Avg HV** | **Avg DU** |
| 4 536 377.9 | 83.0% | 30.67 | 0.335 | 709 014.9 | 94.5% | 30. 46 | 0.405 | 6.4 |

Results from the executed experiments show that with increased dimensions of MCO problems to be solved and a corresponding increase in the volume of calculations, the efficiency of the MAGCS algorithm remains at a high level; its use has efficiently reduced the number of the executed iterations by 6.4 times.

**6. Conclusion**

This paper proposes a new approach for solving computationally expensive lexicographic multicriteria optimization problems (MCOlex), in which the efficiency criteria can be multiextremal, and calculating the criteria values may require a large amount of computation. A key feature of this class of problems is the ability, during the computation process, to alter the ordering of efficiency criteria in terms of importance, which in turn necessitates a multistage solution for MCOlex problems.

Overcoming the enormous computational complexity of addressing the formulated new class of MCOlex problems is ensured by solving a sequence of global optimization problems with nonlinear constraints using efficient information-statistical methods of global optimization by using an original index constraint accounting scheme instead of the commonly used penalty functions. A core element in this approach is the ability to use all the search information obtained in the computation process, utilizing the multistage solution of MCOlex problems. Starting to solve each new stage of the solution, this search information allows us to incorporate the previously calculated values of efficiency criteria into the values of the next scalar problem of multiextremal optimization. The search information retrieved becomes part of the current optimization state and through optimization methods, results in adaptive planning of the iterations to run the global search.

Results of computational experiments demonstrate that this approach can significantly reduce the computational complexity of multistage MCOlex problem solving.

In conclusion, it must be noted that the approach described here is promising, and requires further research. Importantly, it is necessary to continue carrying out computational experiments to solve multicriteria optimization problems with a larger number of efficiency criteria and for greater dimensions in the optimization problems to be solved. It is also necessary to evaluate the possibility of parallel computing using high-performance supercomputer systems.

**Acknowledgements**

This research was supported by the Russian Science Foundation, project No 16-11-10150 “Novel efficient methods and software tools for time-consuming decision making problems using supercomputers of superior performance”.

**References**

1. Miettinen K. (1999) Nonlinear Multiobjective Optimization // Springer.
2. Ehrgott, M. (2005) Multicriteria Optimization // Springer. (2nd ed., 2010)
3. Collette, Y., Siarry, P. (2011) Multiobjective Optimization: Principles and Case Studies (Decision Engineering) // Springer.
4. Marler, R. T., Arora, J. S. (2009). Multi-Objective Optimization: Concepts and Methods for Engineering // VDM Verlag.
5. Pardalos, P.M., Žilinskas, A., Žilinskas, J. (2017) Non-Convex Multi-Objective Optimization. Springer.
6. Marler, R. T., Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering // Struct. Multidisciplinary Optimization 26, 369-395.
7. Figueira,J., Greco, S., Ehrgott, M., editors. (2005). Multiple criteria decision analysis: State of the art surveys // New York (NY): Springer.
8. Zavadskas, E. K., Turskis, Z., Kildienė, S. (2014). State of art surveys of overviews on MCDM/MADM methods// Technological and Economic Development of Economy, 20, 165–179.
9. Hillermeier, C., Jahn, J. (2005). Multiobjective optimization: survey of methods and industrial applications. Surv. Math. Ind. 11, 1–42.
10. Branke, J., Deb, K., Miettinen, K., Slowinski, R., editors. (2008). Multi-Objective Optimization—Interactive and Evolutionary Approaches // Springer, Berlin.
11. Deb, K. (2001). Multi-Objective Optimization using Evolutionary Algorithms // Wiley, Chichester.
12. Yang, X.-S. (2008) Nature-inspired metaheuristic algorithms // Luniver Press, Frome
13. Tan, K.C., Khor, E.F., Lee, T.H. (2005). Multi-objective Evolutionary Algorithms and Applications // Springer-Verlag, London
14. Eichfelder, G. (2009) Scalarizations for adaptively solving multi-objective optimization problems // Comput. Optim. Appl. 44, 249–273
15. Jones, D.R.: A taxonomy of global optimization methods based on response surfaces. J. Glob. Optim. 21, 345–383 (2001)
16. Voutchkov, I., Keane, A.: Multi-objective optimization using surrogates. Comput. Intell. Optim. Adapt. Learn. Optim. 7, 155–175 (2010)
17. Strongin, R.G. (1978) Numerical methods in multiextremal problems: information-statistical algorithms // Nauka, Moscow (in Russian)
18. Strongin, R., Sergeyev, Ya. (2000) Global optimization with non-convex constraints. Sequential and parallel algorithms // Kluwer Academic Publishers, Dordrecht (2nd ed. 2013, 3rd ed. 2014).
19. Törn, A., Žilinskas, A. (1989). Global Optimization. Lecture Notes in Computer Science 350. Berlin: Springer-Verlag.
20. Horst, R., Tuy, H. (1990). Global Optimization: Deterministic Approaches. Berlin: Springer-Verlag.
21. Zhigljavsky, A.A. (1991). Theory of Global Random Search. Dordrecht: Kluwer Academic Publishers.
22. Pintér, J.D. (1996) Global optimization in Action (continuous and Lipschitz optimization: algorithms, implementations and applications) // Kluwer Academic Publishers, Dortrecht.
23. Sergeyev Y.D., Strongin R.G., Lera D. (2013) Introduction to global optimization exploiting space-filling curves // Springer.
24. Locatelli, M., Schoen, F. (2013). Global Optimization: Theory, Algorithms, and Applications. SIAM.
25. Floudas, C.A., Pardalos, M.P. (2016). Recent Advances in Global Optimization. Princeton University Press.
26. Lera D., Sergeyev Ya.D. (2015) Deterministic global optimization using space-filling curves and multiple estimates of Lipschitz and Holder constants, Communications in Nonlinear Science and Numerical Simulation, 23, 328–342.
27. Gergel, V.: A Unified Approach to Use of Coprocessors of Various Types for Solving Global Optimization Problems. 2nd International Conference on Mathematics and Computers in Sciences and in Industry, 13–18 (2015) DOI: 10.1109/MCSI.2015.18
28. Arora, R.K. Optimization: Algorithms and Applications. CRC Press, 2015.
29. Bazaraa, M.S., Sherali, H.D., Shetty, C. M. Nonlinear Programming: Theory and Algorithms, John Wiley and Sons, 2006 (3rd ed.)
30. Gergel, V.P., Kozinov, E.A.: Accelerating multicriterial optimization by the intensive exploitation of accumulated search data. In: AIP Conference Proceedings, 1776, 090003 (2016) DOI: 10.1063/1.4965367
31. Gergel, V.P., Kozinov, E.A.: Efficient multicriterial optimization based on intensive reuse of search information. In: J Glob Optim., 71(1), 73-90 (2018) DOI: 10.1007/s10898-018-0624-3
32. Sysoyev, A., Barkalov, K., Gergel, V. (2018) [Globalizer: A novel supercomputer software system for solving time-consuming global optimization problems](https://www.scopus.com/record/display.uri?eid=2-s2.0-85046437028&origin=resultslist&sort=plf-f&src=s&sid=368c70aa547d795bd51d4b519c04b644&sot=autdocs&sdt=autdocs&sl=17&s=AU-ID%286603540044%29&relpos=24&citeCnt=2&searchTerm=" \o "Показать сведения о документе). [Numerical Algebra, Control and Optimization](https://www.scopus.com/sourceid/21100283718?origin=resultslist" \o "Показать сведения о названии источника) 8(1), с. 47-62
33. Kvasov D.E., Sergeyev Y.D. (2015) Deterministic approaches for solving practical black-box global optimization problems. Advances in Engineering Software, vol. 80, pp.58-66
34. Modorskii, V., Gaynutdinova, D., Gergel, V., Barkalov, K.: Optimization in design of scientfic products for purposes of cavitation problems. AIP Conference Proceedings 1738 (2016)
35. Evtushenko, Yu.G., Posypkin, M.A. (2014). A deterministic algorithm for global multi-objective optimization // Optimization Methods & Software, 29 (5), 1005–1019.
36. Žilinskas, A., Žilinskas, J. (2015). Adaptation of a one-step worst-case optimal univariate algorithm of bi-objective Lipschitz optimization to multidimensional problems // Commun Nonlinear Sci Numer Simulat, 21, 89–98.
37. Gaviano, M., Kvasov, D.E, Lera, D., and Sergeyev, Ya.D.: Software for generation of classes of test functions with known local and global minima for global optimization. ACM Transactions on Mathematical Software 29(4), 469-480 (2003)
38. Zarepisheh M, Khorram E. On the transformation of lexicographic nonlinear multiobjective programs to single objective programs. Math. Methods Oper. Res. 2011;74:217–231.
39. Haimes YY, Lasdon L, Wismer D. On a bicriterion formulation of the problem of integrated systems identification and system optimization. IEEE Trans. Syst. Man Cybern. 1971;1:296–297.
40. Chankong V, Haimes YY. Multiobjective decision making: theory and methodology. Vol. 8, North-Holland Series in System Science and Engineering. New York (NY): Elsevier; 1983.
41. Ehrgott M, Ruzika S. Improved ε-constraint method for multiobjective programming. J. Optim. Theory Appl. 2008;138:375–396.
42. Rastegar, N., Khorram E. Relaxation of constraints in lexicographic multiobjective programming problems. A Journal of Mathematical Programming and Operations Research, 64, 2015, 2111-2129.
43. Castro-Gutierrez J., Landa-Silva D., Pérez J.M. (2010) Improved Dynamic Lexicographic Ordering for Multi-Objective Optimisation. In: Schaefer R., Cotta C., Kołodziej J., Rudolph G. (eds) Parallel Problem Solving from Nature, PPSN XI. PPSN 2010. Lecture Notes in Computer Science, vol 6239. Springer, Berlin, Heidelberg.
44. [Charles Audet](https://epubs.siam.org/author/Audet%2C+Charles), [Gilles Savard](https://epubs.siam.org/author/Savard%2C+Gilles), and [Walid Zghal](https://epubs.siam.org/author/Zghal%2C+Walid). Multiobjective Optimization Through a Series of Single-Objective Formulations SIAM J. Optim., 2008,19(1), 188–210.
45. Yaroslav D. Sergeyev , Dmitri E. Kvasov. Deterministic Global Optimization: An Introduction to the Diagonal Approach (SpringerBriefs in Optimization). 2017 Springer/
46. Carr, C.R., Howe, C.W. . (1964). Quantitative Decision Procedures in Management and Economic: Deterministic Theory and Applications. New York: McGraw–Hill.
47. Dam, E.R., Husslage, B., Hertog, D. (2010). One-dimensional Nested Maximin Designs. Journal of Global Optimization, 46(2), 287–306.
48. V. Gergel, E. Kozinov, E\_cient multicriterial optimization based on intensive reuse of search information, J. Glob. Optim. 71 (1) (2018) 73-90. doi:10.1007/s10898-018-0624-3.
49. L. Martí, J. García, A. Berlanga, J. M. Molina, A stopping criterion for multi-objective optimization evolutionary algorithms, Information Sciences 367 (2016) 700-718.
50. E. Dilettoso, S. A. Rizzo, N. Salerno, A weakly pareto compliant quality indicator, Mathematical and Computational Applications 22 (1) (2017) 25.
51. J. Wu, S. Azarm, Metrics for quality assessment of a multiobjective design optimization solution set, Journal of Mechanical Design 123 (1) (2001) 18-25.
52. C. Audet, J. Bigeon, D. Cartier, S. Le Digabel, L. Salomon. Performance indicators in multiobjective optimization. G-2018-90, 2018. https://www.gerad.ca/en/papers/G-2018-90
53. Huband, S., Hingston, P., Barone, L., and While, L. (2006). A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. IEEE Transactions on Evolutionary Computation, 10(5):477–506.
54. Hansen, N., Auger, A., Finck, S., and Ros, R. (2009). Real-Parameter Black-Box Optimization Benchmarking 2009: Experimental Setup. INRIA Research Report RR-6829, INRIA Saclay—Ile-de-France. updated February 2010.
55. A. L. Custódio, J. F. Aguilar Madeira, Ismael Vaz, Luís N. Vicente. Direct Multisearch for Multiobjective Optimization. IAM Journal on Optimization 21(3), 2011, 1109-1140.
56. Dimo Brockhoff, Tea Tusar, Anne Auger, Nikolaus Hansen. Using Well-Understood Single-Objective Functions in Multiobjective Black-Box Optimization Test Suites. 2019 https://arxiv.org/abs/1604.00359
57. Cococcioni, M., Cudazzo, A., Pappalardo, M., Sergeyev, Y.D. Solving the Lexicographic Multi-Objective Mixed-Integer Linear Programming Problem using branch-and-bound and grossone methodology, Communications in Nonlinear Science and Numerical Simulation, 2020,    84, 105177.
58. Cococcioni, M., Pappalardo, M., Sergeyev, Y.D. Lexicographic multi-objective linear programming using grossone methodology: Theory and algorithm, Applied Mathematics and Computation, 2018,    318, 298-311.

1. Data ordering is reflected by using a subscript. [↑](#footnote-ref-1)
2. If *M*=*m*+1, then is the minimum value of the function ϕ(x). [↑](#footnote-ref-2)
3. This method is also known as the index method - see [18]. [↑](#footnote-ref-3)
4. The values rν>1,1≤ν≤m+1, are the AGCS reliability parameters used for computing the interval characteristics in (25) [↑](#footnote-ref-4)
5. В данной статье в ходе выполнения вычислительных экспериментов проводилось решение до 5000 global optimization problems with nonlinear constraints [↑](#footnote-ref-5)