Computationally Efficient Approach for Solving the Lexicographic Multicriteria Optimization Problems

В.П. Гергель, Е.А. Козинов

Национальный исследовательский Нижегородский государственный университет   
им. Н.И. Лобачевского

В статье предлагается вычислительно эффективный подход для решения сложных лексикографических многокритериальных задач оптимизации, в которых критерии эффективности могут быть многоэкстремальными, а вычисление значений критериев может требовать большого объема вычислений. Предполагается также, что в ходе вычислений постановка задач может меняться и, тем самым, может потребоваться решение динамически определяемых наборов задач многокритериальной оптимизации. Разработанный подход основывается на сведении многомерных задач к одномерным задачам глобальной оптимизации, применении эффективных алгоритмов глобального поиска, разработанных в рамках информационно-статистической теории многоэкстремальной оптимизации, и повторном использовании поисковой информации, получаемой в процессе вычислений. Как показывают результаты вычислительных экспериментов, подобный подход позволяет значительно сократить вычислительную трудоемкость решения задач многокритериальной оптимизации.

*Ключевые слова:*Многокритериальная оптимизация, лексикографическая упорядоченность, глобальная оптимизация, редукция размерности, поисковая информация, вычислительная сложность.

1. Введение

Задачи многокритериальной оптимизации (МКО) относятся к числу наиболее общих постановок для decision-making problems и широко встречаются в приложениях. Многокритериальная оптимизация является областью активных научных исследований, в ходе которых предложено большое количество эффективных методов многокритериальной оптимизации и решено большое множество практических задач - см., например, монографии [1-4] и обзоры научных и практических результатов в данной области [6-9].

Ключевой особенностью задач МКО является отсутствие в большинстве случаев единственного решения (decision), которое было бы наилучшим сразу по всем критериям эффективности в силу их возможной противоречивости. И, как результат, при решении задач МКО обычно требуется найти нескольких компромиссных (effective, non-dominated) decisions, которые не могут быть улучшены без ухудшения эффективности по каким-либо отдельным критериям задачи МКО.

Среди разработанных подходов для решения задач МКО можно выделить методы *лексикографической оптимизации*, когда осуществляется то или иное упорядочивание критериев по важности и оптимизация осуществляется в порядке их упорядоченности – см., например, [3]. Еще один подход представляют *итеративные методы*[6,10], когда в процесс выбора вариантов активно включается исследователь (decision maker). Активно развиваемое направление состоит в разработке и применении для решения задач МКО эволюционных алгоритмов, основанных на имитации тех или иных природных явлений [10-13]. Среди широко используемых направлений для решения задач МКО — подход *скаляризации*, в котором применяются те или способы свертки частных критериевк единому скалярному критерию – см., например, [2,14].

Данная работа посвящена решению задач лексикографической многокритериальной оптимизации (MCOlex), возникающих при проектировании сложных технических объектов и систем. В таких приложениях критерии эффективности могут иметь сложный *многоэкстремальный* вид, а область допустимых вариантов может определяться сложными *немонотонными ограничениями*. Предполагается также, что вычисление значений критериев и ограничений может потребовать *большого объема вычислений*. В таких условиях нахождение даже одного компромиссного decision требует значительных вычислений, определение же нескольких (или всего множества) эффективных decision становится проблемой **большой вычислительной сложности**.

Перечисленные выше свойства выделяют ключевую особенность рассматриваемого класса задач MCOlex – высокая вычислительная сложность. Одно из перспективных направлений для поиска способов решения подобных задач состоит в использовании model-based approach, когда после небольшого количества вычислений значений вычислительно-трудоемких критериев и ограничений строятся быстро-вычислимые аппроксимирующие функции [15-16]. Такой подход является достаточно эффективным, однако построение хороших аппроксимаций является затруднительным при многоэкстремальном поведении оптимизируемых критериев и ограничений.

Предлагаемый в работе подход для решения вычислительно-трудоемкого класса задач MCOlex основывается на следующих основных положениях. Прежде всего, решение задачи сводится к решению последовательности задач глобальной оптимизации с нелинейными ограничениями (GCO) [2,14]. Затем, для решения задач GСO применяется эффективные алгоритмы глобального поиска, разработанные в рамках информационно-статистической теории многоэкстремальной оптимизации [17-18]. И, наконец, при выполнении всех необходимых вычислений полностью используется вся поисковая информация, получаемая в процессе решения задачи MCOlex. В целом, разработанный подход позволяет существенно уменьшить объема выполняемых вычислений вплоть до выполнения всего лишь нескольких итераций при поиске очередных эффективных decisions.

Дальнейшая структура статьи имеет следующий вид. В главе 2 новый класс задач оптимизации - задач многоэтапной многокритериальной лексикографической оптимизации (MMCOlex) – решение которых сводится к решению последовательности задач глобальной оптимизации с нелинейными ограничениями. В главе 3 представлены основы разработанного подхода: сведение многомерных MCOlex задач к одномерным задачам глобальной оптимизации, применение эффективных алгоритмов глобального поиска, разработанных в рамках информационно-статистической теории многоэкстремальной оптимизации, и повторное использование поисковой информации, получаемой в процессе вычислений. Глава 4 содержит результаты численных экспериментов. В заключении обсуждаются полученные результаты и приводятся основные направления продолжения исследований.

# 2. Постановка задачи

Задача многокритериальной (или векторной) оптимизации (MCO) может быть определена следующим образом:

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

где *f*(*y*) = (*f*1(*y*), *f*2(*y*),…, *f*s(*y*)) есть векторный критерий эффективности, *y* = (*y*1, *y*2,…, *yN*) есть вектор варьируемых параметров, а *N* есть размерность решаемой задачи многокритериальной оптимизации. Область поиска *D* задает множество возможных значений параметров и обычно представляет собой *N*-мерный гиперкуб

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

при заданных граничных векторах *a* и *b*.

Не уменьшая общности, предполагается, что значения критериев эффективности не отрицательны и их уменьшение соответствует повышению эффективности рассматриваемых вариантов *y*∈*D.* Задача (1) в данной работе рассматривается применительно к наиболее сложным проблемам принятия решений, в которых критерии эффективности *fi*(*y*), 1≤*i*≤*s,* могут быть существенно многоэкстремальными, а процедуры вычисления значений критериев в точках области поиска *y*∈*D* могут быть вычислительно трудоемкими. Предполагается также, что критерии *fi*(*y*), 1≤*i*≤*s,* удовлетворяют условию Липщица

|  |  |
| --- | --- |
| , 1≤*i*≤*s,* | (2) |

where *Li* is the Lipschitz constant for the function *fi*(*y*), 1≤*i*≤*s* and  denotes the Euclidean norm in . Выполнимость условия Липщица означает, что при небольших вариациях параметра *y*∈*D* соответствующие изменения значений критериев *fi*(*y*), 1≤*i*≤*s,* являются ограниченными.

В задаче (1) критерии эффективности обычно противоречивы и не существует варианта *y*∈*D*, который обеспечивал бы наилучшие (наименьшие) значения для всех критериев одновременно, т.е.

|  |  |
| --- | --- |
| *.* | (3) |

В случае справедливости соотношения (3) при решении задачи МКО определяется такие варианты ∈*D*, значения критериев в которых не могут быть улучшены без ухудшения показателей эффективности по каким-либо отдельным критериям *fi*(*y*), 1≤*i*≤*s,* т.е.

|  |  |
| --- | --- |
| *y*'∈*D* : (*fi*(*y*')≤*fi* (), 1≤*i*≤*s*), ( : *fj*(*y*')<*fj* ()). | (4) |

Подобные неулучшаемые decisions называются *эффективными* или *оптимальными* по Парето.

Множество Парето *P*(*f*,*D*) может оказаться достаточно большим, что усложняет возможность анализа эффективных вариантов для лица, принимающего решения. Возможный способ сокращения числа рассматриваемых эффективных вариантов состоит в предположении упорядоченности критериев эффективности по важности, что часто имеет место в практических предположениях.

Не уменьшая общности, будем предполагать далее упорядоченность критериев *fi*(*y*), 1≤*i*≤*s* по важности в соответствии с их перечислением в (1). Упорядоченность критериев определяет отношение линейного порядка в области поиска *D*, т.е.

|  |  |
| --- | --- |
| . | (4) |

В результате задача (1) сводится к задаче многокритериальной лексикографической оптимизации (MCOlex)

|  |  |
| --- | --- |
| *f*(*y*) = (*f*1(*y*), *f*2(*y*),…, *f*s(*y*)) → minlex, *y*∈*D*, | (5) |

решение которой осуществляется поэтапно: сначала выполняется оптимизация первого (наиболее важного) критерия; далее, если решение является не единственным, то осуществляется оптимизация второго критерия на множестве решений первого этапа и т.д.

|  |  |
| --- | --- |
| , | (6) |

где обозначение *Arg* означает множество всех вариантов (points), при которых достигается минимальное значение оптимизируемого критерия. Следует отметить, что последовательность этапов в (6) может быть выполнена не полностью, если на каком-то этапе множество вырождается и содержит только единственное decision *у*∈*D*.

Получаемое в результате (6) множество решений *Plex*(*f*,*D*) является подмножеством области Парето *P*(*f*,*D*) и может содержать от одного до нескольких вариантов (points) ∈*D* в случае, когда минимальное значение последнего критерия достигается в нескольких точках области поиска *D*. Такое резкое сокращение множества рассматриваемых эффективных решений может оказаться нежелательным. Расширение множества решений *Plex*(*f*,*D*) может быть обеспечено за счет некоторого ослабления «строгого» лексикографического порядка и разрешать на каждом этапе вычислительной схемы (6) выбор вариантов в некоторой окрестности минимальных значений оптимизируемого критерия, т.е.

|  |  |
| --- | --- |
|  | (7) |

Такой подход широко известен также как метод последовательных уступок (the method of successive concessions (MSC)) [3-5]. Выбор уступок *δi*, 1≤*i*≤*s* в вычислительной схеме (7) позволяет получать любые эффективные решения ∈*D* и учитывать особенности решаемой задачи MCOlex. Вместе с тем, такой поход приводит к необходимости решения на каждом этапе схемы (7) более сложных задач глобальной оптимизации с нелинейными ограничениями на всей области поиска *D*.

Следует отметить также, что в процессе вычисления может потребоваться изменение выбранных значений для уступок *δi*, 1≤*i*≤*s* – уступки могут оказаться достаточно жесткими или, наоборот, чрезмерно большими. В самом общем случае, может потребоваться и изменение порядка важности критериев эффективности. Учет подобных предположение приводит к необходимости более общей постановки процесса решения задачи MCOlex и обеспечить возможность решения множества задач вида (7)

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

который может изменяться динамически в ходе вычислений путем добавления новых или удаления уже существующих задач многокритериальной оптимизации. В целом, подобный подход определяет новый класс задач оптимизации - задач многоэтапной лексикографической многокритериальной оптимизации (MMCOlex).

# 3. The Approach: Key Ideas and Methods

Предложенная расширенная формулировка задач многокритериальной лексикографической оптимизации (7)-(8) предполагает многократное решение задач многоэкстремальной оптимизации с нелинейными ограничениями. Задачи данного вида являются вычислительно трудоемкими и подвержены «проклятию размерности» - вычислительная сложность экспоненциально увеличивается с ростом размерности. Предлагаемый в статье подход для решения подобных задач основывается на трех ключевых идеях:

* Редукция многомерных задач многоэкстремальной оптимизации к задачам одномерного глобального поиска с использованием отображений на основе кривых Пеано,
* Применение индексного метода глобальной оптимизации, позволяющего решать задачи оптимизации с ограничениями без использования штрафных функций,
* Повторное использование поисковой информации, получаемой в процессе вычислений, для многократном решении задач глобальной оптимизации.

## 3.1. Редукция размерности для многомерных задач многокритериальной оптимизации

Поиск численных оценок для глобально оптимальных точек предполагает построение сеток, покрывающих область поиска D - см., например, [17-25]. Данная ключевая особенность задач глобальной оптимизации приводит к «проклятию размерности» - вычислительная сложность построения покрытий экспоненциально возрастает при увеличении размерности решаемой задачи оптимизации. Значительное снижение сложности может быть достигнуто применения неоднородных сеток, которые являются более плотными только в окрестности глобально оптимальных решений. Подобные неоднородные покрытия области поиска D могут быть построены адаптивно, когда при выборе точек очередных итераций глобального поиска выполняется с учетом всей имеющейся поисковой информацией (точки предыдущих итераций поиска и значения оптимизируемой функции в этих точках), т.е.

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

где *k* есть номер выполняемой итерации глобального поиска, есть решающее правило применяемого алгоритма глобального поиска, в соответствии с которым выполняется выбор точки очередной итерации, а есть поисковая информация, имеющаяся на *k* шаге глобального поиска.

Адаптивный выбор точек на выполняемых итерациях глобального поиска предполагает анализ объемной многомерной поисковой информации (, что определяет большую вычислительную сложность решающих правил алгоритмов глобального поиска. Данная вычислительная сложность может существенно снижена за счет редукции решаемых задач оптимизации с использованием *кривых* или *разверток* Пеано *y*(*x*), однозначно и непрерывно отображающих отрезок [0,1] на N-мерный гиперкуб D– см., например, [17-18,23]. В результате такой редукции исходная многомерная задача многокритериальной оптимизации (5) сводится к одномерной задаче:

|  |  |
| --- | --- |
| *f*(*y*(*x*)) = (*f*1(*y*(*x*)), *f*2(*y*(*x*)),…, *f*s(*y*(*x*))) → minlex, x∈[0,1].. | (10) |

Следует отметить, что получаемые в результате редукции одномерные функции удовлетворяют равномерному условию Гельдера (см. [17-18]), т.е.

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

где константы *Hi* определяется соотношением ),1≤*i*≤*m,* *Li* есть константы Липшица из (2), а *N* есть размерность задачи оптимизации (1).

Применение редукции размерности позволяет привести поисковую информацию из (9) к виду

|  |  |
| --- | --- |
| , | (12) |

в котором в дополнение к редуцированному представлению данные расположены в порядке возрастания[[1]](#footnote-1) точек , т.е.

|  |  |
| --- | --- |
|  | (12) |

для более эффективного выполнения алгоритмов глобального поиска.

Редукция размерности с использованием кривых Пеано широко используется при разработке эффективных алгоритмов глобального поиска в рамках информационно-статической теории многоэкстремальной оптимизации [17,18,23,26-27]. Численные методы для построения аппроксимаций кривых Пеано (*разверток*) с заданной точностью для заданной размерности рассмотрены в [17-18].

## 3.2. Эффективный метод решения задач глобальной оптимизации с нелинейными ограничениями

Представим задачи, решаемые в рамках вычислительной схемы (7), в виде одномерных задач глобальной оптимизации с нелинейными ограничениями

|  |  |
| --- | --- |
| min {ϕ(x) : gi(x)≤0, 1≤*i*≤*m*,x∈=[0,1] }. | (13) |

(далее будет использоваться также обозначение gm+1(x) = ϕ(x)).

Наличие нелинейных ограничений заметно усложняет трудоемкость решения задач глобальной оптимизации – получаемые решения должны принадлежать допустимой области

|  |  |
| --- | --- |
| . | (14) |

Для снижения сложности решения задач оптимизации с ограничениями часто выделяют более простые случаи – например, выделяются задачи с линейными или квадратичными ограничениями. Широко применяются различные способы аппроксимации сложных ограничений при помощи ограничений более простого вида (линейных, выпуклых и т.п.). Но наиболее часто используется метод штрафных функций, когда при вычислениях в недопустимой области Δ\ к значениям минимизируемой функции добавляется некоторый штраф. Однако, метод штрафных функций часто предполагает многократное решение задачи оптимизации для определения достаточного значения коэффициента штрафа. Более полная информация о методах решения оптимизационных задач с ограничениями может быть получена, например, в монографиях [18,26-27].

Используемый в работе подход основывается на оригинальном методе раздельного учета ограничений, разработанного в рамках информационно-статистической теории глобального поиска [18]. Суть подхода состоит в построении некоторой интегральной целевой функции без ограничений, решение которой обеспечивает также и решение исходной задачи (13) – более подробное изложение данного метода приводится далее.

Применим последовательную схему для получения значений ограничений gi(y(x))≤0, 1≤*i*≤*m*, и минимизируемой функции ϕ(x), в соответствии с которой значения ограничений вычисляются строго по порядку их номеров, и вычисления сразу же прекращаются при обнаружении первого нарушенного ограничения (схема частичной вычислимости). Количество ограничений, в которых вычислялись значения, будет именоваться далее как *индекс* ν=ν(x), т.е.

|  |  |
| --- | --- |
| ν(x) : gν(x)>0, gj(x)≤0, 1≤j≤ν−1, 1≤ν=ν(x)≤m | (15) |

(при выполнении всех ограничений полагается ν=m+1). Введенное понятие индекса позволяет определить для задачи (13) интегрированную функцию (см. рис. 1)

|  |  |
| --- | --- |
| F(x) = gν(x), ν=ν(x), | (16) |

определенную и вычислимую всюду на [0,1]*.* Ее значение в точке x есть либо значение левой части ограничения, нарушенного в этой точке (случай, когда ν ≤ m), либо значение минимизируемой функции (случай, когда ν=m+1). Процесс вычисления значения F(x), x∈[0,1], завершается либо в результате установления неравенства gj(x)>0, либо в результате достижения значения ν(x)=m+1 (данная процедура называется далее *испытанием*).

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

а) б)

Рис. 1. Интегрированная функция для задачи (13) при использовании последовательной схемы вычисления ограничений: а) функция *F*(*x*) из (16), б) функция из (17)

Функция F(x) состоит из отдельных фрагментов ограничений gi(x)≤0, 1≤*i*≤*m*, и минимизируемой функции ϕ(x). Для однородности составных частей далее функция F(x) преобразуется к виду

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

где *M* есть номер ограничения, для которого отсутствуют допустимые точки (максимальное значение индекса ν(x)),  есть величина минимального нарушения ограничения [[2]](#footnote-2), а ,1≤ν≤*M*, есть константы Гельдера из (11). Как правило, данные величины а приори неизвестны, но при выполнении вычислений вместо этих величин можно использовать их адаптивные оценки, получаемые в процессе решения задачи на основании результатов выполненных испытаний. Для этого поисковая информация из (12) должна быть расширена данными

|  |  |
| --- | --- |
| , | (18) |

где . Тогда оценки требуемых величин могут быть определены следующим образом

|  |  |
| --- | --- |
| Mk = max {ν=ν(xi), 1≤i≤k }, | (19) |
|  | (20) |
| . | (21) |

Если величина в (21) оказывается неопределенной или равна нулю, то принимается =1. Для начальных значений данных величин может быть принято M0 = =1, =0, .

Общая вычислительная схема обобщенного *алгоритма глобального поиска* для задач многоэкстремальной оптимизации с нелинейными ограничениями[[3]](#footnote-3) (AGCS) состоит в следующем [18].

Первое испытание осуществляется в произвольной внутренней точке x1∈(0,1)*.* Результаты вычислений постоянно добавляются в множество из (12); кроме того, в дополнительно вводятся точки x0=0 и xk+1=1 (значения z0 и zk+1 не определены и не используются) для удобства последующих обозначений. Выбор точки xk+1, k≥1, любого последующего испытания определяется следующими правилами.

*Правило*1. Для каждого интервала (xi−1,xi), 1≤i≤k+1, в множестве вычислить *характеристику* R(i), где

|  |  |
| --- | --- |
|  | (22) |
| . |

Величины rν>1*,* 1≤ν≤m+1, являются параметрами алгоритма. Произведения rνhν используется как оценку констант Гельдера Hν, 1≤ν≤m+1 из (11).

*Правило*2. Определить интервал (xt−1,xt), которому соответствует максимальная характеристика

|  |  |
| --- | --- |
| R(t)=max{R(i): 1≤i≤k+1}. | (23) |

*Правило*3. Провести очередное испытание в точке интервала *x*k+1∈(xt−1,xt), определяемой в соответствии с выражением

|  |  |
| --- | --- |
|  | (24) |

Итерации алгоритма прекращаются, если выполняется условие остановки

|  |  |
| --- | --- |
| ρt≤ε, | (25) |

где t из (19) и ε>0 есть заданная точность.

Результаты применения алгоритма AGCS для решения тестовой задачи из рис. 1 с параметрами r1=r2=r3=2 и ε=10−5показаны на рис. 2. Координаты точек испытаний, осуществленных алгоритмом в процессе решения задачи, отмечены на рис. 2 тремя рядами вертикальных штрихов. Штрихи верхнего ряда соответствуют точкам с единичным индексом, второго – точкам, индексы которых равны 2; точки, отмеченные штрихами нижнего ряда, являются допустимыми. Координаты испытаний, выполненных в близких точках, отмечены темным прямоугольником. Всего значение первого ограничения вычислялось 147 раз, второго ограничения – 84 раза, а значение минимизируемой функции вычислялось всего лишь 35 раз.

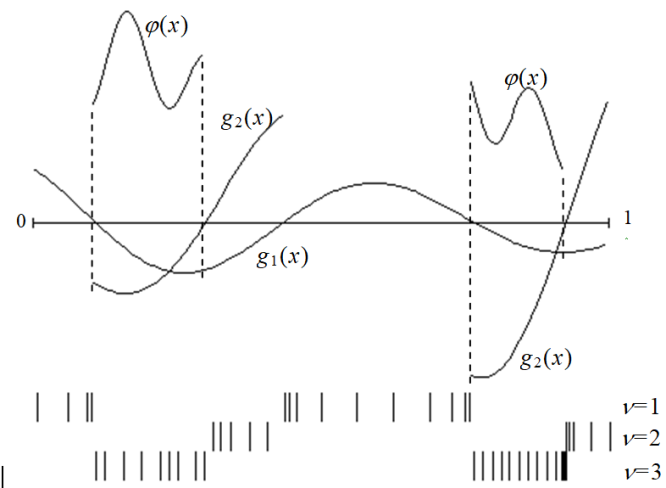


Рис. 2. Результаты применения алгоритма AGCS для решения тестовой задачи

Various modifications of this algorithm and the corresponding theory of convergence are presented in [18]. В данной работе показана справедливость следующей теоремы.

**Теорема**. Если алгоритм AGCS применяется для решения задачи (13), и при этом выполняются условия:

1. каждая функция gj, 1≤j≤m+1,  [0,1] удовлетворяет условию Гельдера из (11) с константой Hj ,
2. для величин hν из (21), начиная с некоторого шага, справедливы неравенства

|  |  |
| --- | --- |
| rνhν>2Hν, 1≤ν≤m+1, |  |

то множество предельных точек последовательности {xk}, порождаемой алгоритмом, совпадет с множеством решений задачи (13) при ε=0 в условии остановки (25).

## 3.3. Ускорение вычислений на основе повторного использования поисковой информации

Как уже отмечалось ранее, задача MCOlex на каждом этапе метода последовательных уступок (7) представляет собой задачу глобальной оптимизации с нелинейными ограничениями (13). В общем случае, решение каждой такой задачи должно начинаться с самого начала, и подобное многократное решение задач глобальной оптимизации может потребовать большого объема вычислений. Однако наличие поисковой информации из (18) позволяет привести результаты предыдущих вычислений к значениям любой очередной решаемой задачи оптимизации схемы (7) без каких-либо трудоемких вычислений значений критериев исходнойзадачи (1) [30-31]. Подобное повторное использование поисковой информации позволит сократить объем вычислений для решения каждой последующей оптимизационной задачи вплоть до выполнения всего лишь нескольких итераций для поиска очередного эффективного варианта (см. Раздел 4 с результатами вычислительных экспериментов).

Более того, использование поисковой информации позволяет свести многоэтапное решение задачи МКО в соответствии со схемой (7) к решению единственной задаче глобальной оптимизации последнего этапа схемы (7), а именно

|  |  |
| --- | --- |
|  | (26) |

где для неизвестных а приори величин можно использовать оценки из (20), получаемые на основе поисковой информации , а для уступок можно использовать значения, нормированные к диапазону изменения ограничений, т.е.

|  |  |
| --- | --- |
| . | (27) |

Применимость подобной одноэтапной схемы (26)-(27)по сравнению со схемой (7) оценивается в Разделе 4 при проведении вычислительных экспериментов.

При полной схеме вычисления критериев, т.е. когда в точках всех выполненных итерациях глобального поиска вычислены значения всех критериев эффект от использования поисковой информации может оказаться еще более значительным. В этом случае, поисковая информация может использоваться и при изменении значений уступок и, тем самым, решение очередной задачи семейства из (8) может осуществляться каждый раз с использованием результатов всех ранее выполненных вычислений. Как показали результаты вычислительных экспериментов (см. Раздел 4) сокращение объема выполняемых вычислений за счет повторного использования поисковой информации – не менее чем в 6.4 раз.

Алгоритм AGCS, дополненный возможностью использования поисковой информации при решении множества задач MCOlex из (7), будет именоваться далее алгоритмом глобального поиска для многоэтапного решения множества задач многоэкстремальной оптимизации с нелинейными ограничениями (MAGCS).

# 4. Результаты вычислительных экспериментов

Вычислительные эксперименты проводились на суперкомпьютере «Лобачевский» Нижегородского государственного университета (операционная система – CentOS 6.4, система управления – SLURM). Один узел суперкомпьютера располагает 2-я процессорами Intel Sandy Bridge E5-2660 2.2 GHz, 64 Gb RAM. Центральный процессор является 8-и ядерным (т.е. всего на узле доступно 16 ядер CPU). Для получения исполняемого программного кода использовался компилятор Intel C++ 14.0.2. The numerical experiments were performed using the Globalizer system [32].

Алгоритмы многоэкстремальной оптимизации, используемые в рамках разработанного подхода, показали свою эффективность при проведении тестовых вычислительных экспериментов и широко применялись при решении практических задач глобального поиска – см., например, [33-34]. Далее приводятся результаты вычислительных экспериментов при решении задач многокритериальной оптимизации.

В первой серии вычислительных экспериментов было проведено сравнение разработанного алгоритма MAGCS c несколькими алгоритмами многокритериальной оптимизации. Для сравнения использовалась тестовая двухкритериальная задача, предложенная в [35]:

(28)

Под решением задачи МКО понималось построение численной аппроксимации области Парето. Для оценки качества аппроксимации сравнивались полнота и равномерность покрытия области Парето с помощью следующих двух показателей [35-36]:

* The hypervolume index (HV). Данный показатель характеризует полноту аппроксимации области Парето (большее значение соответствует более полному покрытию области Парето).
* The distribution uniformity index (DU). Данный показатель характеризует равномерность покрытия области Парето (меньшее значение соответствует более равномерному покрытию области Парето).

В рамках данного эксперимента сравнивались пять алгоритмов многокритериальной оптимизации: 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], и алгоритм MACGS, предложенного в данной статье. Результаты решения задачи (29) для всех перечисленных методов (кроме MAGCS) были получены в [36].

Для MAGСS было решено 50 задач (3) при разных значениях величин , равномерно распределенных в интервале [0,1]. Использовались точность метода из (25) и надежность из (22) . В полном виде результаты выполненных экспериментов представлены в таблице 1.

Таблица 1. Сравнение эффективности алгоритмов многокритериальной оптимизации

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Метод решения** | MC | SEMO | NUC | BLO | **MAGCS** |
| **Количество  итераций метода** | 500 | 500 | 515 | 498 | **273** |
| **Количество найденных точек области Парето** | 67 | 104 | 29 | 68 | **80** |
| **HV индекс** | 0.300 | 0.312 | 0.306 | 0.308 | **0.314** |
| **DU индекс** | 1.277 | 1.116 | 0.210 | 0.175 | **0.096** |

Как показывают результаты выполненных экспериментов, алгоритм MAGСS имеет заметное преимущество по сравнению с рассмотренными методами многокритериальной оптимизации даже при решении сравнительно простых задач МКО.

Во второй серии вычислительных экспериментов производилось решение двухкритериальных двумерных задач МКО. В качестве критериев задачи использовались многоэкстремальные функции, получаемые при помощи генератора GKLS [37]. В ходе экспериментов было выполнено решение 100 многокритериальных задач данного класса. В каждой задаче производился поиск Парето-оптимальных вариантов для 50 разных значений величин , равномерно распределенных в интервале [0,1] (т.е. всего было решено 5000 задач глобальной оптимизации). Получаемые результаты усреднялись по количеству решенных задач МКО.

В вычислительных экспериментах поиск решения производился с остановкой по достижению точности метода. В конце решения контролировалась правильность найденного решения. Для контроля сравнивалась точки решения, найденного методом и точки из Парето-границы, вычисленной с учетом выбранных значений величин. При решении серии задач использовалась точность метода , параметры метода . Результаты вычислительных экспериментов представлены в таблице 2.

В таблице 2 в первом и пятом столбцах указано среднее количество итераций, затрачиваемых алгоритмом MAGCS на решение задач MCOlex. Второй и шестой столбцы содержат процент решенных задач при заданных параметрах метода. В третьем, четвертом седьмом и восьмом столбцах приведены значения показателей HV и DU. Последний столбец показывает величину сокращения количества выполняемых итераций глобального поиска при решении задач MCOlex за счет повторного использования поисковой информации.

Таблица 2. Результаты серии экспериментов по решению   
двумерных двухкритериальных МКО задач

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Поисковая информация** | | | | | | | | **Сокращение количества итераций** |
| **не используется** | | | | **используется** | | | |
| **Итераций  метода** | **Решенных задач** | **Avg HV** | **Avg DU** | **Итераций  метода** | **Решенных задач** | **Avg HV** | **Avg DU** |
| 41 710.1 | 98.0% | 33.43 | 0.173 | 2 407.41 | 99.3% | 33.34 | 0.224 | 17.3 |

Полученные результаты экспериментов показывают, что повторное использование поисковой информации позволяет сократить общий объем вычислений в 17.3 раз без использования дополнительных вычислительных ресурсов, при этом согласно усредненным показателям HV и DU качество найденной области Парето остается в среднем на том же уровне.

В третьей серии вычислительных экспериментов производилось решение двухкритериальных четырехмерных задач МКО. Как и во второй серии вычислительных экспериментов, решалось 100 многокритериальных задач. В каждой задаче производился поиск Парето-оптимальных вариантов для 50 разных значений величин , равномерно распределенных в интервале [0,1]. Критерии решаемых задач МКО определялись при помощи генератора GKLS [37]. При решении серии задач использовалась точность метода , параметры метода . Результаты вычислительных экспериментов представлены в таблице 3.

Таблица 3. Результаты серии экспериментов по решению   
четырехмерных двухкритериальных МКО

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Поисковая информация** | | | | | | | | **Сокращение количества итераций** |
| **не используется** | | | | **используется** | | | |
| **Итераций  метода** | **Решенных задач** | **Avg HV** | **Avg DU** | **Итераций  метода** | **Решенных задач** | **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 |

Результаты экспериментов показывают, что при увеличении размерности решаемых задач МКО и соответствующего увеличения объема вычислений эффективность алгоритма МAGCS остается на высоком уровне – достигаемое сокращение количества выполняемых итераций глобального поиска составляет 6.4 раз.

# 6. Заключение

В статье предлагается новый подход для решения вычислительно сложных лексикографических многокритериальных задач оптимизации (MCOlex), в которых критерии оптимальности могут быть многоэкстремальными, а вычисление значений критериев может требовать большого объема вычислений. Ключевой особенностью рассматриваемого класса задач является возможность изменения упорядоченности критериев эффективности по важности в процесс вычислений, что приводит к необходимости многоэтапного решения MCOlex задач.

Преодоление большой вычислительной сложности решения сформулированного нового класса MCOlex задач обеспечивается посредством решения последовательности задач глобальной оптимизации с нелинейными ограничениями с помощью эффективных информационно-статистических методов глобальной оптимизации с оригинальной индексной схемы учета ограничений вместо обычно используемых штрафных функций. Ключевым элементом разработанного подхода является возможность использования всей поисковой информации, получаемой в процессе вычислений, при многоэтапном решении MCOlex задач л. Наличие поисковой информации позволяет при переходе к новому этапу решения приводить вычисленные ранее значения критериев эффективности к значениям очередной решаемой скалярной задачи многоэкстремальной оптимизации. Приведенная к актуальному состоянию поисковая информация используется методами оптимизации для адаптивного планирования выполняемых итераций глобального поиска.

Как показывают результаты вычислительных экспериментов, разработанный подход позволяет значительно сократить вычислительную трудоемкость многоэтапного решения MCOlex задач.

Как заключение, можно отметить, что разработанный подход является перспективным и требует дальнейшего продолжения исследований. Прежде всего, необходимо продолжить проведение вычислительных экспериментов по решению задач многокритериальной оптимизации при большем количестве частных критериев эффективности и для большей размерности решаемых задач оптимизации. Следует также оценить возможность организации параллельных вычислений с использованием высокопроизводительных суперкомпьютерных систем.

# 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. X2. Miettinen K. (1999) Nonlinear Multiobjective Optimization // Springer.
2. X3. Ehrgott, M. (2005) Multicriteria Optimization // Springer. (2nd ed., 2010)
3. X4. Collette, Y., Siarry, P. (2011) Multiobjective Optimization: Principles and Case Studies (Decision Engineering) // Springer.
4. X34. 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. Х6. Marler, R. T., Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering // Struct. Multidisciplinary Optimization 26, 369-395.
7. Х7. Figueira,J., Greco, S., Ehrgott, M., editors. (2005). Multiple criteria decision analysis: State of the art surveys // New York (NY): Springer.
8. X9. 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. X35. Hillermeier, C., Jahn, J. (2005). Multiobjective optimization: survey of methods and industrial applications. Surv. Math. Ind. 11, 1–42.
10. X32. Branke, J., Deb, K., Miettinen, K., Slowinski, R., editors. (2008). Multi-Objective Optimization—Interactive and Evolutionary Approaches // Springer, Berlin.
11. X37. Deb, K. (2001). Multi-Objective Optimization using Evolutionary Algorithms // Wiley, Chichester.
12. X33. Yang, X.-S. (2008) Nature-inspired metaheuristic algorithms // Luniver Press, Frome
13. X38. Tan, K.C., Khor, E.F., Lee, T.H. (2005). Multi-objective Evolutionary Algorithms and Applications // Springer-Verlag, London
14. X8. Eichfelder, G. (2009) Scalarizations for adaptively solving multi-objective optimization problems // Comput. Optim. Appl. 44, 249–273
15. X40. Jones, D.R.: A taxonomy of global optimization methods based on response surfaces. J. Glob. Optim. 21, 345–383 (2001)
16. X41. Voutchkov, I., Keane, A.: Multi-objective optimization using surrogates. Comput. Intell. Optim. Adapt. Learn. Optim. 7, 155–175 (2010)
17. X12. Strongin, R.G. (1978) Numerical methods in multiextremal problems: information-statistical algorithms // Nauka, Moscow (in Russian)
18. X13. 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. X28. Törn, A., Žilinskas, A. (1989). Global Optimization. Lecture Notes in Computer Science 350. Berlin: Springer-Verlag.
20. X22. Horst, R., Tuy, H. (1990). Global Optimization: Deterministic Approaches. Berlin: Springer-Verlag.
21. X29. Zhigljavsky, A.A. (1991). Theory of Global Random Search. Dordrecht: Kluwer Academic Publishers.
22. X11. Pintér, J.D. (1996) Global optimization in Action (continuous and Lipschitz optimization: algorithms, implementations and applications) // Kluwer Academic Publishers, Dortrecht.
23. X14. Sergeyev Y.D., Strongin R.G., Lera D. (2013) Introduction to global optimization exploiting space-filling curves // Springer.
24. X23. Locatelli, M., Schoen, F. (2013). Global Optimization: Theory, Algorithms, and Applications. SIAM.
25. X16. 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. X42. Arora, R.K. Optimization: Algorithms and Applications. CRC Press, 2015.
29. X43. 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=). [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. X50. 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. X36. Evtushenko, Yu.G., Posypkin, M.A. (2014). A deterministic algorithm for global multi-objective optimization // Optimization Methods & Software, 29 (5), 1005–1019.
36. X39. 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. X44. 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)

1. Упорядоченность данных отражается использованием нижнего индекса [↑](#footnote-ref-1)
2. Если *M*=*m*+1, то есть минимальное значение функции ϕ(x). [↑](#footnote-ref-2)
3. Данный алгоритм известен также как индексный метод - см. [X13]. [↑](#footnote-ref-3)