

УДК 519.85

ВИКОМПОЗИЦИЯ КОНУСНЫХ ПРОЕКЦИЙ¹**Е. А. Нурминский**

Рассматривается декомпозиционный подход для решения задачи ортогонального проектирования заданной точки на выпуклый полиэдральный конус, заданный конечным множеством своих образующих. Сводимость проблемы линейной оптимизации к подобной проекционной задаче потенциально делает этот подход одним из возможных новых способов решения задач линейного программирования большой размерности. В качестве основы подобного подхода может быть предложена идея рекуррентной бинарной декомпозиции, представляющей исходную задачу большой размерности в виде бинарного дерева конусных проекций последовательного разложения первоначального конуса на сумму конусов меньших размерностей. Базовой операцией такого подхода является решение задачи проекции некоторой точки на конус, представленный как сумма двух подконусов при минимальной, по возможности, их модификации и достаточно произвольном их выборе. В работе предложены три итеративных алгоритма, реализующие эту базовую операцию, доказана их сходимость, проведены вычислительные эксперименты, демонстрирующие как вычислительную эффективность предлагаемого подхода, так и некоторые проблемы, возникающие при его применении.

Ключевые слова: ортогональная проекция, полиэдральные конусы, декомпозиция, линейная оптимизация

E. A. Nurminski. A bicomposition of conical projections.

We consider a decompositional approach to the problem of finding the orthogonal projection of a given point onto a convex polyhedral cone represented by a finite set of its generators. The reducibility of an arbitrary linear optimization problem to such projection problem potentially makes this approach one of the possible new ways to solve large-scale linear programming problems. Such an approach can be based on the idea of a recurrent binary decomposition that splits the original large-scale problem into a binary tree of conical projections corresponding to a sequential decomposition of the initial cone into the sum of lesser subcones. The key operation of this approach is solving the problem of projecting of a certain point onto a cone represented as the sum of two subcones with the smallest possible modification of these subcones and their arbitrary choice. Three iterative algorithms implementing this basic operation are proposed, their convergence is proved, and numerical experiments demonstrating both the computational efficiency of the algorithms and certain problems of their application are performed.

Keywords: orthogonal projection, polyhedral cones, decomposition, linear optimization.

MSC: 47H09, 90C25, 90C06

DOI: 10.21538/0134-4889-2023-29-3-73-87

Введение

Задачи линейной оптимизации, даже в их классической постановке, остаются важным элементом математических ядер систем анализа и принятия решений в цифровой трансформации. В этой области накоплен колоссальный опыт решения практических задач, проведено большое количество экспериментов по поиску эффективных вычислительных стратегий, реализовано значительное число солверов промышленного уровня, среди которых, по данным популярного сетевого оптимизационного сервера NEOS [1], в области линейной оптимизации с большим отрывом лидирует CPLEX [2]. В числе ближайших конкурентов можно упомянуть

¹Работа выполнена в Дальневосточном центре математических исследований при финансовой поддержке Минобрнауки России (соглашение № 075-02-2023-946 от 16 февраля 2023 г. по реализации программ развития региональных научно-образовательных математических центров).

GUROBI [3] и недавно появившийся, но быстро набирающий популярность COPT [4]. Между этими продуктами идет ожесточенная конкурентная борьба, и различные исследования по разным классам задач дают разный рейтинг этих солверов, но их доминирование позволяет говорить об олигополистическом характере данного рынка. Среди программных продуктов с открытым кодом для решения задач линейной оптимизации наиболее популярным является GLPK [5], предоставляющий помимо готового оптимизирующего солвера еще и богатый набор различных функций обработки данных, связанных с задачами линейной оптимизации.

По широко распространяемым данным сравнительных испытаний, проводимых изготовителями коммерческих солверов, их продукты существенно превосходят программы с открытым кодом (см. например [6]), что вполне закономерно, учитывая масштаб работ, проделанных над проприетарным программным обеспечением. Кроме того, зачастую высокая эффективность проприетарных солверов связана с привязкой к определенным компьютерным архитектурам, моделям процессоров или другой аппаратной составляющей, доступ к информации о которой также ограничен.

Однако предпочтение высокоэффективных проприетарных продуктов и ориентация только на них имеют ряд негативных последствий. Закрытый характер этого программного обеспечения, по сути дела, препятствует развитию оптимизации как научного направления, создавая чрезмерно высокий входной порог для конкурирующих подходов. Источники эффективности коммерческих кодов скрыты условиями предоставления, произвольно выставляемыми правообладателями, что затрудняет для пользователей разработку более совершенных моделей. Практика лицензирования проприетарного программного обеспечения в условиях значительной монополизации рынка ставит пользователя в зависимость от неконтролируемых им факторов, что может оказать негативное влияние на его производственные процессы. Прекращение действия лицензии основного продукта в этом случае означает либо (в лучшем случае) замораживание развития систем, которые его используют, либо затратную замену на сходную систему, либо существенную переработку всех компонент, связанных с этим продуктом (такая переработка может понадобиться, например, для нейтрализации рисков, обнаруженных в замороженной версии); в худшем случае производитель может просто лишить пользователя возможности использовать его продукт.

Это достаточный набор аргументов для того, чтобы продолжать открытое развитие алгоритмики и программного обеспечения для решения задач линейной оптимизации, и в данной работе продолжено развитие проекционного подхода к ее решению. Основой этого подхода является показанная в [7] возможность решить задачу линейной оптимизации с помощью единичной ортогональной проекции на допустимое множество данной задачи. В случае линейной оптимизации эта задача может иметь весьма высокую размерность, представлять собой полуопределенную задачу квадратичного программирования, и ее решение в общем случае становится определенной проблемой. Поэтому весьма желательно развивать для нее специализированные алгоритмы понижения размерности, в частности основанные на принципе декомпозиции.

Алгоритмы, предложенные в настоящей работе, основаны на сведении задачи проекции на полиэдр к задаче проекции на конус, порождаемый образующими, которые соответствуют линейным неравенствам, описывающим полиэдр [8]. Группируя эти образующие в различные объединения, можно свести процесс решения исходной проблемы к проекциям на подконусы, соответствующие этим объединениям и помимо эффекта понижения размерности использовать возможности крупнозернистой параллелизации. Основное значение при этом приобретает базовая подзадача нахождения проекции на конус при помощи разбиения его на два подконуса и получения решения исходной проекционной задачи в результате итеративного процесса проектирования на подконусы и координации этих проекций. Чтобы подчеркнуть бинарный характер такого подхода, мы будем называть его бикомпозицией.

1. Обозначения и предварительные результаты

Как правило, используются стандартные обозначения конечномерного выпуклого анализа, но тем не менее они приведены здесь во избежание путаницы и неправильного понимания.

Основное евклидово векторное пространство, в котором рассматривается задача проекции, обозначается как E или E^n , если его размерность n имеет значение. Элементы этого пространства, или векторы, обозначаются, как правило, строчными латинскими буквами — a, b, \dots, x, y, z — при необходимости с нижними и верхними индексами, диакритическими маркерами и пр. Мы используем специальные обозначения для нулевого вектора $\mathbf{0} = (0, 0, \dots, 0)$ и вектора из единиц $\mathbf{1} = (1, 1, \dots, 1)$.

Неравенства между векторами, алгебраические суммы векторов и умножение векторов на вещественные числа определяются покомпонентно. Последние две операции естественным образом обобщаются на множества векторов: $A + B = \{a + b, a \in A, b \in B\}$, $\alpha A = \{\alpha a, a \in A\}$. Линейное подпространство $L \subset E$ определяется как множество, инвариантное по отношению к сложению и умножению на вещественные числа: $L = L + L = \alpha L$ для любого вещественного $\alpha \neq 0$.

Неотрицательная часть E обозначается как $E_+ = \{x \geq \mathbf{0}\}$.

Множество, состоящее из одного элемента, скажем a , будем в плане обозначений отождествлять с самим элементом.

Выпуклая оболочка множества X определяется и обозначается общепринятым образом:

$$\text{co}(X) = \left\{ z = \sum_{i=1}^{|X_f|} \alpha_i x^i, x^i \in X_f \subset X, |X_f| < \infty, \alpha_i \geq 0, \sum_{i=1}^{|X_f|} \alpha_i = 1 \right\},$$

где $|X_f|$ — кардинальность конечного подмножества X_f множества X . Безусловно, вследствие теоремы Каратеодори в конечномерном пространстве E^n достаточно рассмотреть множества X_f с $|X_f| \leq n + 1$ [9]. Для экономии обозначений мы определим коническую оболочку множества X как $X^< = \bigcup_{\alpha \geq 0} \alpha \text{co}(X)$.

Скалярное произведение векторов x и y из E обозначается через xy , и норма вектора x — это $\|x\|^2 = xx$. Норма множества $A \subset E$ определяется как $\|A\| = \sup_{a \in A} \|a\|$. Используя скалярное произведение, ортогональное дополнение L^\perp линейного подпространства $L \subset E$ определяем как $L^\perp = \{x: xz = 0 \text{ для всех } z \in L\}$. Среди других специальных множеств отметим канонический симплекс: $\Delta = E_+ \cap \{x: \mathbf{1}x = 1\}$.

Операция проекции точки p на выпуклое замкнутое множество X в E определена и обозначается как

$$p \downarrow X = \operatorname{argmin}_{x \in X} \|p - x\|,$$

т. е. $p \downarrow X \in X$, $\min_{x \in X} \|p - x\| = \|p - p \downarrow X\|$. Для замкнутых выпуклых X эта операция хорошо определена и липшицево непрерывна относительно p с константой Липшица, не превосходящей 1, т. е. $\|p \downarrow X - q \downarrow X\| \leq \|p - q\|$. Эта операция естественным образом обобщается на множества: $A \downarrow X = \{a \downarrow X, a \in A\} \subset X$.

Краткий список алгебраических свойств проекции:

- идемпотентность: $(p \downarrow X) \downarrow X = p \downarrow X$;
- линейность: для линейных подпространств $L \subset E$

$$(\alpha p) \downarrow L = \alpha(p \downarrow L) \text{ для } \alpha \in \mathbb{R}, \quad (p + q) \downarrow L = p \downarrow L + q \downarrow L;$$

- операции сдвига и масштабирования сочетаются с проекцией следующим образом:

$$(a + b) \downarrow X = a \downarrow (X - b) + b, \quad (\tau a) \downarrow X = \tau(a \downarrow (\tau^{-1} X)) \text{ для } \tau > 0;$$

- ортогональная декомпозиция: $p = p \downarrow L + p \downarrow L^\perp$;
- инвариантность по отношению к ортогональным сдвигам: так как $L^\perp \downarrow L = L \downarrow L^\perp = \mathbf{0}$, то $X \downarrow L = (X + L_\perp) \downarrow L$ для любого линейного подпространства $L \subset E$.

Для замкнутого ограниченного множества $X \subset E$ и вектора c определим и обозначим опорную функцию $(X)_c = \sup_{x \in X} cx = cx^c$, где $x^c \in X$. Это выпуклая суплинейная (иногда используется термин суперлинейная) функция своего аргумента c , надграфик которой представляет собой выпуклый конус с вершиной в начале координат. Образующие конуса порождаются элементами множества X , и функция линейна на каждом луче, выходящем из начала координат.

Как уже упоминалось, основой сведения задачи линейной оптимизации к проекционной задаче служит следующее утверждение.

Теорема 1 ([7], теорема 1). Пусть задача линейной оптимизации $\min_{x \in X} cx$ с ограниченным полиэдральным множеством $X = \{x: Ax \leq b\}$ имеет единственное решение x^* . Тогда для произвольного x^0 существует такое $\tau > 0$, что $(x^0 - \tau c) \downarrow X = x^*$.

Эта теорема в ослабленной формулировке ($x^* = (x^0 - \tau c) \downarrow X$ является одним из решений задачи линейной оптимизации) справедлива и для исходных задач линейной оптимизации с неединственным решением.

В свою очередь задача вычисления проекции $(x^0 - \tau c) \downarrow X = p \downarrow X$ с помощью теории двойственности сводится (см. [8] и сильно упрощенную версию [10]) к вычислению $e \downarrow K$, где e — некоторая фиксированная точка, которая не зависит от задачи, а K — выпуклый полиэдральный конус, порожденный исходными данными задачи линейной оптимизации. Слега упрощая, можно говорить о том, что образующие конуса K соответствуют строкам матрицы ограничений. В представленной работе мы уделяем внимание именно последней задаче, считая конус K заданным своим конечным набором образующих.

В заключение приведем простую лемму, которая легко доказывается от противного.

Лемма 1. Пусть $\{\alpha_k\}$ и $\{\beta_k\}$ — две монотонно убывающие неотрицательные последовательности, такие, что $\max\{\alpha_{k+1}, \beta_{k+1}\} \leq \min\{\alpha_k, \beta_k\}$. Тогда данные последовательности имеют общий предел $\lim_{k \rightarrow \infty} \alpha_k = \lim_{k \rightarrow \infty} \beta_k$,

Эта лемма существенно используется для доказательства сходимости алгоритма 2 — параллелизуемого метода бикомпозиции с единственным координирующим вектором.

2. Алгоритмы конической бикомпозиции задачи линейной оптимизации

Имея в перспективе приложения в области линейной оптимизации, мы рассмотрим полиэдральный конус K , заданный как коническая оболочка некоторого конечного множества образующих $G = \{g^1, g^2, \dots, g^m\}$, т. е.

$$K = G^< = \left\{ x = \sum_{i=1}^m \lambda_i g^i, \lambda_i \geq 0, i = 1, 2, \dots, m \right\}.$$

В соответствии с этим исходная задача проекции, к которой сводится задача линейной оптимизации принимает вид

$$p \downarrow K = \operatorname{argmin}_{z \in K} \|p - z\|^2, \quad (2.1)$$

где $p = (0, 0, \dots, 0, 1)$, а элементы множества G определяются строками ограничений задачи линейной оптимизации $\min_{Ax \leq b} cx$ и вектором стоимостей c . В данной работе мы будем считать множество G заданным, а подробности этого процесса описаны в [7; 8].

Бикомпозицию конуса K будем понимать как его представление в виде суммы двух конусов K_A и K_B , которые индуцируются разбиением множества G на два подмножества

$$G = G_A \cup G_B, \quad G_A = \{g^i, i \in I_A\}, \quad G_B = \{g^i, i \in I_B\},$$

где I_A и I_B в совокупности покрывают весь интервал индексов образующих: $\{1, 2, \dots, m\} = I_A \cup I_B$. Другими словами, $K = K_A + K_B = \{K_A \cup K_B\}^<$, где $K_A = G_A^<$, $K_B = G_B^<$. Индексные подмножества I_A и I_B могут и пересекаться, что не сказывается на дальнейших результатах.

Проективная задача (2.1) преобразуется при этом в эквивалентную:

$$\min_{z \in K_A + K_B} \|p - z\|^2 = \min_{\substack{z_A \in K_A \\ z_B \in K_B}} \|p - z_A - z_B\|^2 = \min_{\substack{z_A \in K_A \\ z_B \in K_B}} \rho(z_A, z_B) \quad (2.2)$$

— с разделяемыми ограничениями $z_A \in K_A$ и $z_B \in K_B$ и предоставляет возможность использовать методы типа координатного спуска [11] или последовательных проекций, особенно, если какой-то из конусов K_A, K_B или оба имеют простую структуру вида координатных ортантов или порождаются простыми, например, сильно разреженными, образующими.

В последующих разделах описаны три итеративных алгоритма проекции, использующие несколько отличающиеся варианты применения специфики задачи (2.2). В описаниях алгоритмов допущена некоторая вольность в том плане, что алгоритмы производят в принципе бесконечно число итераций, и символический выход из цикла означает, что предельные точки генерируемой последовательности решают задачу проекции.

2.1. Метод последовательных проекций

Одна из первых идей, которую, на наш взгляд, стоит рассмотреть, — это применение схемы покоординатного спуска, использованной еще Дж. фон Нейманом в его классической работе (см., например, [12]) о нахождении кратчайшего расстояния между двумя выпуклыми множествами.

Используя этот подход, можно предложить алгоритм 1, сходимость которого утверждает теорема 2.

Algorithm 1. Концептуальная схема алгоритма последовательного проектирования

Data: Конусы K_A, K_B , такие, что $K = K_A + K_B$, вектор $p \notin K$

Result: Точка $z^* = p \downarrow K$

Инициализация;

Задать произвольные начальные приближения $z_A^0 \in K_A$, $z_B^0 \in K_B$;

Установить счетчик итераций $k = 0$;

while $z_A^k \neq z_B^k$ **do**

$$\rho(z_A^{k+1}, z_B^k) = \min_{z_A \in K_A} \rho(z_A, z_B^k) = \min_{z_A \in K_A} \|p - z_B^k - z_A\|; \quad (2.3)$$

$$\rho(z_A^{k+1}, z_B^{k+1}) = \min_{z_B \in K_B} \rho(z_A^{k+1}, z_B) = \min_{z_B \in K_B} \|p - z_A^{k+1} - z_B\|; \quad (2.4)$$

Увеличить счетчик итераций $k \leftarrow k + 1$;

end

Теорема 2. Если конус K не содержит нетривиального линейного подпространства, то последовательность $\{z^k = z_A^k + z_B^k\}$, сгенерированная алгоритмом 1, при $k \rightarrow \infty$ сходится к решению задачи (2.2).

Доказательство. По построению $\|p - z_B^k - z_A^{k+1}\| \leq \|p - z_B^k - z_A^k\|$ для любого $z_A \in K_A$, и, следовательно,

$$\|p - z_B^k - z_A^{k+1}\| \leq \|p - z_B^k - z_A^k\|. \quad (2.5)$$

Аналогично по построению $\|p - z_B^{k+1} - z_A^{k+1}\| \leq \|p - z_B^k - z_A^{k+1}\|$ для любого $z_B \in K_B$, значит,

$$\|p - z_B^{k+1} - z_A^{k+1}\| \leq \|p - z_B^k - z_A^{k+1}\|. \quad (2.6)$$

Складывая (2.5) и (2.6), получаем

$$\|p - z_B^{k+1} - z_A^{k+1}\| + \|p - z_B^k - z_A^{k+1}\| \leq \|p - z_B^k - z_A^{k+1}\| + \|p - z_B^k - z_A^k\|$$

или $\|p - z_B^{k+1} - z_A^{k+1}\| \leq \|p - z_B^k - z_A^k\|$. Отсюда, конечно, следует, что у последовательности $\{\|p - z_B^k - z_A^k\|, k = 1, 2, \dots\}$ есть предел

$$\rho = \lim_{k \rightarrow \infty} \|p - z_B^k - z_A^k\|$$

и последовательность $z_B^k + z_A^k \in K_A + K_B = K$, $k = 1, 2, \dots$ с $z_A^k \in K_A$, $z_B^k \in K_B$ является равномерно ограниченной. По предположению теоремы конус K не содержит нетривиального линейного подпространства, последовательности $\{z_A^k\}, \{z_B^k\}, k = 1, 2, \dots$ ограничены и, значит, у них есть предельные точки $z_A^* \in K_A, z_B^* \in K_B$. Поскольку в силу (2.5) и (2.6)

$$\|p - z_B^{k+1} - z_A^{k+1}\| \leq \|p - z_B^k - z_A^{k+1}\| \leq \|p - z_B^k - z_A^k\|,$$

тот же предел имеет и $\|p - z_B^k - z_A^{k+1}\| \rightarrow \rho$ при $k \rightarrow \infty$.

Рассматривая возникающие при этом предельные точки z_B, z_A, z_A^+ последовательностей $\{z_B^k\}, \{z_A^k\}, \{z_A^{k+1}\}$, можно утверждать, что $\rho = \|p_B - z_A^+\| = \min_{z \in K_A} \|p_B - z\| = \|p_B - z_A\| = \rho$, где $p_B = p - z_B$. При этом мы воспользовались единственностью и равномерной непрерывностью решения типовой задачи проекции $\min_{z \in K} \|q - z\|$ по q , которые гарантируют, что

$$\lim_{q \rightarrow \bar{q}} \min_{z \in K} \|q - z\| = \min_{z \in K} \|\bar{q} - z\|$$

для произвольных \bar{q} и выпуклых конусов K .

Учитывая, что $z_A \in K_A$ и что решение задачи $\min_{z \in K_A} \|p_B - z\|$ единственно, получаем отношение $z_A = z_A^+$ или доказательство того, что $\{z_A^k\}, k = 1, 2, \dots$, имеет единственную предельную точку, которую, как и ранее, будем обозначать через z_A^* . Аналогично показывается, что $\{z_B^k\}, k = 1, 2, \dots$, также имеет единственную предельную точку, которую будем обозначать через z_B^* . Очевидно, что $z_A^* \in K_A, z_B^* \in K_B$.

Тогда в силу предельных условий оптимальности в задачах (2.3), (2.4) справедливы неравенства

$$(p - z_A^* - z_B^*)z_B \leq 0, \quad (p - z_A^* - z_B^*)z_A \leq 0 \quad \text{для любых } z_A \in K_A, z_B \in K_B,$$

сложив которые получим для $z^* = z_A^* + z_B^* \in K = K_A + K_B$

$$(p - z^*)z \leq 0 \quad \text{для любых } z \in K,$$

что гарантирует $z^* = p \downarrow K$.

Интересно отметить, что, хотя алгоритм последовательных проекций может рассматриваться как вариант метода покоординатного спуска, задача (2.2) не удовлетворяет ни одному из стандартных условий сходимости этого метода, таких как сильная выпуклость, дифференцируемость целевой функции, отсутствие непокоординатных ограничений (см., например, [11; 13; 14]). Тем не менее согласно доказанному алгоритм 1 сходится. \square

2.2. Параллелизуемые алгоритмы минимальной координации

Основное вычислительное преимущество, которое можно ожидать от декомпозиции задач большого объема — это параллелизация вычислений, которая дает возможность уменьшить объем данных, обрабатываемых разделенными вычислительными устройствами, и сократить время вычислений на итерацию.

Типично время вычислений в наших экспериментах с решением генерируемых общих задач линейной оптимизации без определенной специфики типа сетевых, транспортных задач, росло нелинейно, примерно как куб объема данных. Соответственно уменьшение объема данных при бикомпозиции в 2 раза вызывает при полной параллелизации уменьшение времени вычислений, как минимум, в 8 раз. В соответствии с этим можно надеяться на то, что этот эффект перекроет дополнительные вычислительные затраты на координацию решений подзадач.

2.2.1. Алгоритм симметричной бикомпозиции с единственным координирующим вектором. В надежде на такую экономию концептуальная схема простейшего алгоритма симметричной бикомпозиции с единственным координирующим вектором для проекции точки p на конус $K = \{K_A, K_B\}^<$ может быть описана алгоритмом 2. Конечно, в практических вычислениях условие равенства $z_A = z_B$ следует заменить на условие достаточной малости разности $z_A - z_B$.

Algorithm 2. Концептуальная схема алгоритма симметричной бикомпозиции с единственным координирующим вектором

Data: Конусы K_A, K_B , такие, что $K = K_A + K_B$, вектор $p \notin K$

Result: Точка $z^* = p \downarrow K$

Инициализация;

Задать произвольное начальное приближение $q^0 \in K$;

Установить счетчик итераций $k = 0$;

while $z_A^k \neq z_B^k$ **do**

$$z_A^k = p \downarrow \{K_A, q^k\}^<; \quad z_B^k = p \downarrow \{K_B, q^k\}^<; \quad (2.7)$$

 Переопределить координирующий вектор;

$$q^{k+1} = p \downarrow \{z_A^k, z_B^k\}^<; \quad (2.8)$$

 Увеличить счетчик итераций $k \leftarrow k + 1$;

end

Очевидными преимуществами этой схемы являются независимость проекций для определения вкладов подконусов K_A, K_B в итоговую проекцию, минимальная модификация конусов K_A, K_B в задачах проекции (2.7) и простейшая координирующая задача (2.8) для получения решения задачи (2.1).

Сходимость этого алгоритма утверждается следующей теоремой.

Теорема 3. Последовательности $\{z_A^k, z_B^k\}$, сгенерированные алгоритмом 2, при $k \rightarrow \infty$ сходятся к решению задачи (2.2).

Доказательство. Для упрощения выкладок введем обозначения $K_A^k = \{K_A, q^k\}^<$ и $K_B^k = \{K_B, q^k\}^<$. Обозначим также $p \downarrow K_A^k = p_A^k$, $p \downarrow K_B^k = p_B^k$ и $\|p_A^k - p\| = \alpha_k$, $\|p_B^k - p\| = \beta_k$.

Пусть $B^k = \{p_A^k, p_B^k\}^<$. Тогда согласно (2.8) $q^k = p \downarrow B^k$. Так как p_A^k и p_B^k принадлежат B^k , то

$$\|q^k - p\| \leq \|p_A^k - p\| = \alpha_k \quad \text{и} \quad \|q^k - p\| \leq \|p_B^k - p\| = \beta_k$$

или

$$\|q^k - p\| \leq \min\{\alpha_k, \beta_k\}. \quad (2.9)$$

С другой стороны,

$$\|q^k - p\| = \|p \downarrow B^k - p\| \geq \|p \downarrow \{K_A^k, B^k\}^< - p\| = \|p \downarrow K_A^{k+1} - p\| = \alpha_{k+1}$$

здесь мы использовали тот легко проверяемый факт, что поскольку $B^k = \{z = \lambda_A p_A^k + \lambda_B p_B^k; \lambda_A, \lambda_B \geq 0\}$, то

$$\{K_A^k, \lambda_A p_A^k + \lambda_B p_B^k\}^< = \{K_A^k, p_B^k\}^< = K_A^{k+1}.$$

Аналогично показывается, что

$$\|q^k - p\| \geq \|p \downarrow (\{K_B^k, B^k\}^<) - p\| = \|p \downarrow K_B^{k+1} - p\| = \beta_{k+1};$$

это в совокупности дает

$$\|q^k - p\| \geq \max\{\alpha_{k+1}, \beta_{k+1}\}. \quad (2.10)$$

Сопоставляя (2.9) и (2.10), получаем $\max\{\alpha_{k+1}, \beta_{k+1}\} \leq \min\{\alpha_k, \beta_k\}$, что в силу леммы 1 доказывает

$$\lim_{k \rightarrow \infty} \alpha_k = \lim_{k \rightarrow \infty} \|p \downarrow K_A^k - p\| = \lim_{k \rightarrow \infty} \|p \downarrow K_B^k - p\| = \lim_{k \rightarrow \infty} \beta_k = \rho.$$

Учитывая ограниченность шара $S = \{x: \|x - p\| \leq 2\rho\}$ и то, что при достаточно больших k точки $p \downarrow K_A^k, p \downarrow K_B^k, q^n$ принадлежат S , можно считать, что последовательности $\{p \downarrow K_A^k, p \downarrow K_B^k, q^k\}$ сходятся к некоторым $p_A^\infty, p_B^\infty, q^\infty$ соответственно, причем $\|p_A^\infty - p\| = \|p_B^\infty - p\| = \|q^\infty - p\| = \rho$. Если $p_A^\infty \neq p_B^\infty$, то из простой геометрии следует, что

$$\|p^\infty - p\| < \rho,$$

где мы обозначили $p^\infty = \frac{1}{2}(p_A^\infty + p_B^\infty)$.

Действительно, шар $B = \{z: \|z - p\|^2 \leq \rho^2\}$ представляет собой лебегово множество $\{z: \beta(z) \leq \rho^2\}$ сильно выпуклой функции $\beta(z) = \|z - p\|^2$. Соответственно

$$\beta\left(\frac{p_A^\infty + p_B^\infty}{2}\right) = \beta(p^\infty) < \frac{\beta(p_A^\infty) + \beta(p_B^\infty)}{2} = \rho^2.$$

Вводя (замыкания) пределы Куратовского K_A^∞, K_B^∞ последовательностей $\{K_A^k\}, \{K_B^k\}$, построенных по правилу (2.7), получаем

$$K_A^\infty = \{K_A^\infty, p_B^\infty\}^<, \quad K_B^\infty = \{K_B^\infty, p_A^\infty\}^<;$$

из этого следует, что $p_A^\infty \in K_B^\infty$ и $p_B^\infty \in K_A^\infty$. Поскольку по построению $p_A^\infty \in K_A^\infty$ и $p_B^\infty \in K_B^\infty$, то $p_A^\infty \in K_A^\infty \cap K_B^\infty$ и $p_B^\infty \in K_A^\infty \cap K_B^\infty$; откуда следует, что $p^\infty \in K_A^\infty \cap K_B^\infty$.

Тогда получаем следующее противоречие:

$$\rho > \|p^\infty - p\| \geq \|p_A^\infty - p\|,$$

что доказывает $p_A^\infty = p_B^\infty$. В силу произвольности выбора сходящихся подпоследовательностей $\{p \downarrow K_A^k, p \downarrow K_B^k\}$ отсюда следует равенство их пределов вне зависимости от их выбора и, значит, $p_A^\infty = p_B^\infty = q^\infty$.

При этом $q^\infty = p \downarrow \{K_A \cup q^\infty\}^< = p \downarrow \{K_B \cup q^\infty\}^<$ и из условий оптимальности

$$(p - q^\infty)z_A \leq 0, \quad (p - q^\infty)z_B \leq 0 \quad \text{для } z_A \in \{K_A \cup q^\infty\}^<, \quad z_B \in \{K_B \cup q^\infty\}^<$$

Поскольку $K_A \subset \{K_A \cup q^\infty\}^<$ и $K_B \subset \{K_B \cup q^\infty\}^<$, то и

$$(p - q^\infty)z_A \leq 0 \quad \text{для } z_A \in K_A, \quad (2.11)$$

$$(p - q^\infty)z_B \leq 0 \text{ для } z_B \in K_B. \quad (2.12)$$

Складывая (2.11) и (2.12), получаем

$$(p - q^\infty)(z_A + z_B) \leq 0 \text{ для произвольных } z_A \in K_A, z_B \in K_B$$

или

$$(p - q^\infty)z \leq 0 \text{ для } z \in K_A + K_B = K,$$

что доказывает $q^\infty = p \downarrow K$.

2.2.2. Алгоритм симметричной бикомпозиции с двойным обменом. Очевидно, что при агрегировании координирующей информации в один вектор происходит некоторая потеря информации и для более детального представления процесса сведения проекций можно предложить процедуру симметричной бикомпозиции с двойным обменом.

Как и ранее, мы рассматриваем разложение исходного конуса K задачи (2.2) на пару конусов $K_A = G_A^<$, $K_B = G_B^<$, где G_A и G_B — подмножества образующих, в совокупности покрывающих множество образующих G конуса $K = G^<$, т.е. $G = G_A \cup G_B$. Очевидно, что при этом возникает соответствующая декомпозиция конуса K :

$$K = (K_A \cup K_B)^< = K_A + K_B, \text{ где } K_A = G_A^<, K_B = G_B^<.$$

Концептуальная схема алгоритма симметричной бикомпозиции с двойным обменом представлена в алгоритме 3. Конечно, в практических вычислениях условие равенства $z_A = z_B$ следует заменить на условие достаточной малости разности $z_A - z_B$.

Algorithm 3. Концептуальная схема алгоритма симметричной бикомпозиции с двойным обменом

Data: Конусы K_A, K_B , такие, что $K = K_A + K_B$, вектор $p \notin K$

Result: Вектор $z^* = p \downarrow K$

Инициализация;

$z_A^0 = p \downarrow K_A, z_B^0 = p \downarrow K_B$;

Установить счетчик итераций $k = 0$;

while $z_A^k \neq z_B^k$ **do**

$z_A^{k+1} = p \downarrow \{K_A, z_A^k, z_B^k\}^<$; $z_B^{k+1} = p \downarrow \{K_B, z_A^k, z_B^k\}^<$;
 Увеличить счетчик итераций $k \leftarrow k + 1$;

end

Сходимость алгоритма перекрестных обменов подтверждает следующая теорема.

Теорема 4. Алгоритм 3 генерирует последовательности $\{z_A^k, z_B^k\}$, сходящиеся к решению задачи (2.2).

Доказательство. По построению

$$\|p_1^{k+1} - p\| \leq \min\{\|p \downarrow K_1 - p\|, \|p \downarrow (p_2^k)^< - p\|\} \leq \|p \downarrow (p_2^k)^< - p\| \leq \|p_2^k - p\| \quad (2.13)$$

и

$$\|p_2^{k+1} - p\| \leq \min\{\|p \downarrow K_2 - p\|, \|p \downarrow (p_1^k)^< - p\|\} \leq \|p \downarrow (p_1^k)^< - p\| \leq \|p_1^k - p\|. \quad (2.14)$$

Из (2.13), (2.14) следует, что $\|p_1^{k+1} - p\| \leq \|p_2^k - p\| \leq \|p_1^{k-1} - p\|$, и, симметрично,

$$\|p_2^{k+1} - p\| \leq \|p_1^k - p\| \leq \|p_2^{k-1} - p\|,$$

т.е. $v_{k,1} = \|p_1^k - p\|$, $k = 1, 2, \dots$, и $v_{k,2} = \|p_2^k - p\|$, $k = 1, 2, \dots$, образуют неотрицательные невозрастающие последовательности сходящиеся к некоторым пределам $v_1 = \lim_{k \rightarrow \infty} v_{k,1}$,

$v_2 = \lim_{k \rightarrow \infty} v_{k,2}$. Переходя в (2.13), (2.14) к пределу по $k \rightarrow \infty$, получаем $v_1 \leq v_2 \leq v_1$; откуда следует, что $v_1 = v_2 = v$.

В силу ограниченности последовательностей $\{\bar{p}_1^k\}$ и $\{\bar{p}_2^k\}$ их можно, не умаляя общности, считать сходящимися:

$$\lim_{k \rightarrow \infty} \bar{p}_1^k = \bar{p}_1 \quad \text{и} \quad \lim_{k \rightarrow \infty} \bar{p}_2^k = \bar{p}_2$$

с $\|\bar{p}_1 - p\| = \|\bar{p}_2 - p\| = v$.

Предположим теперь, что $\bar{p}_1 \neq \bar{p}_2$ и рассмотрим $\bar{p} = \frac{1}{2}\bar{p}_1 + \frac{1}{2}\bar{p}_2$. В силу строгой выпуклости единичной сферы

$$\bar{p} \in \text{int}(\{z : \|z - p\| \leq v = \frac{1}{2}\|\bar{p}_1 - p\| + \frac{1}{2}\|\bar{p}_2 - p\|\}),$$

и, следовательно, $\|\bar{p} - p\| < v$. С другой стороны, по построению

$$\bar{p} = \frac{1}{2}\bar{p}_1 + \frac{1}{2}\bar{p}_2 \in \bar{p}_1^< + K_2 = (\bar{p}_1 \cup K_2)^<$$

с $\bar{p}_1 = \lim_{k \rightarrow \infty} p_1^k$, и, значит, $\lim_{k \rightarrow \infty} p \downarrow (p_1^k + K_2)^< = p \downarrow (\bar{p}_1 + K_2)^<$ по непрерывности. Тогда $\|p - \bar{p}\| \geq \|p \downarrow (p_1^k + K_2) - p\| - \epsilon$ для любого $\epsilon > 0$ и достаточно больших k , что приводит к следующему противоречию:

$$\|\bar{p} - p\| < \|\bar{p}_2 - p\| = \|\bar{p}_1 - p\| \leq \|\bar{p} - p\|,$$

и, следовательно, $\bar{p}_1 = \bar{p}_2$. □

3. Демонстрационные расчеты

В качестве тестовых задач рассматривались проекции на случайные конусы с достаточно разнообразными геометрическими характеристиками образующих и различным соотношением размерности пространства и количества образующих. В каждом тестовом наборе данных случайные векторы, которые являются образующими конуса, генерировались в прямоугольном брус $B \subset E^n$ с помощью масштабируемых соответствующим образом случайных чисел, равномерно распределенных на интервале $[0, 1]$. Этот брус можно описать системой неравенств

$$B = \{-L \leq x_i \leq L, i = 1, 2, \dots, n-1; 0 < h \leq x_n \leq h + \delta\}.$$

Вектор p , проектируемый на конус, порожденный этими образующими, определялся как $p = (p_1, p_2, \dots, p_{n-1}, 0)$, где $p_i, i = 1, 2, \dots, n-1$ являлись независимыми случайными величинами, равномерно распределенными на интервале $[-1, 1]$. Очевидно, в силу того что последняя координата p равна нулю, $p \notin B$ и задача нахождения $p^B = p \downarrow B^<$ не является тривиальной. Варьируя параметры L, h, δ , можно получить различные по сложности задачи. Соотношение между параметрами L, h, δ влияют на операции линейной алгебры, используемые при вычислении ядра проекционного оператора в проекционном алгоритме и пересчете базиса в традиционном алгоритме линейной оптимизации. При $h, \delta \rightarrow +0$ и фиксированном $L > 0$ система образующих как правило вырождается и возникает плохо обусловленная задача проектирования точки на близко расположенное линейное подпространство или переобращении базисной матрицы в симплекс-методе.

На рис. 1–5 показаны результаты тестов, проведенных с описанными алгоритмами.

3.1. Сходимость алгоритма последовательного проектирования

Эксперименты с методом последовательного проектирования ожидаемо показали весьма медленную сходимость. Типичным примером является эксперимент с пространством размерности 28, в котором был сгенерирован случайный конус с 58 образующими. В качестве показателя сходимости алгоритма использовалось относительное отклонение суммы проекций на подконусы от истинной проекции на исходный конус, которая вычислялась отдельно.

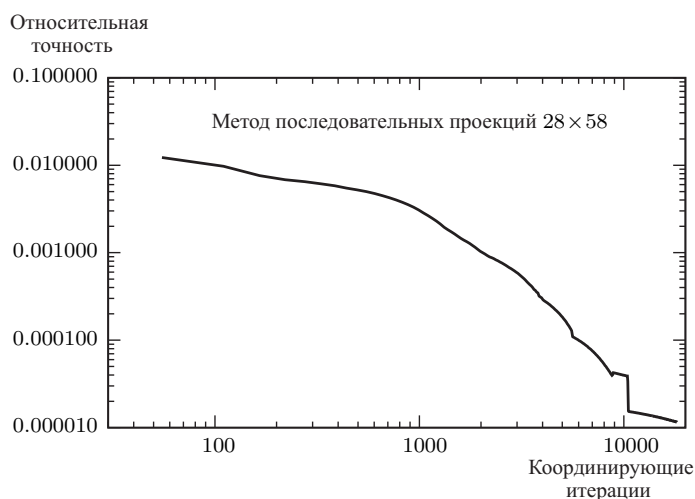


Рис. 1. Метод последовательного проектирования.

Результаты, приведенные на рис. 1, не выявляют определенного характера сходимости. С некоторой долей уверенности его можно определить как сходимость порядка $O(k^{-1/3})$, с некоторым ускорением в финальной фазе вычислений, что не создает для этого метода особых перспектив. В связи с такими показателями сходимости эксперименты с этим методом далее в рамках данной работы не производились.

3.2. Методы параллельного проектирования

Интерес представляют, конечно, случаи достаточно большой размерности и это создает в задаче проекции дополнительные проблемы. Как легко понять, при достаточно большой размерности случайные точки будут распределены практически по поверхности бруса B , точнее по его граням большой размерности и точка p будет иметь весьма небольшие шансы оказаться вблизи какой-либо из образующих. Таким образом, определение p^B потребует достаточно большого объема вычислений для определения подмножества активных образующих и соответствующих барицентрических координат.

Для балансировки подзадач в алгоритмах 2, 3 исходный конус разбивался на два подконуса, образованные примерно одним и тем же количеством образующих. Как показано в теоремах 3, 4, при использовании указанных методов обе проекции на эти конусы, дополненные координирующей информацией, сходятся к общей точке, которая является решением задачи проекции на полный конус. Поэтому сходимость нормы разности этих векторов к нулю является естественным индикатором скорости сходимости и использовалась в качестве критерия оптимальности полученного решения.

3.2.1. Сходимость проекционного алгоритма с единственным координирующим вектором. Интересным результатом этих экспериментов являлась замечательная устойчивость параметров сходимости как по размерности пространства, так и по соотношениям между размерностью и количеством образующих (см. рис. 2, 3). По этим данным скорость метода на первых 10 итерациях можно оценить как устойчивую оценку порядка $O(k^{-4})$, что, конечно, достаточно неплохо. При скорости сходимости $O(k^{-4})$ относительная точность решения становится порядка 10^{-4} после 10 итераций, что вполне удовлетворительно для практических задач линейной оптимизации. Промышленные пакеты линейной оптимизации обычно используют критерии останова порядка 10^{-5} . При скорости сходимости $O(k^{-3})$ для этого достаточно 20 итераций. Вместе с тем после 10-й итерации наблюдается драматическое замедление сходимости, что требует дальнейшего анализа.

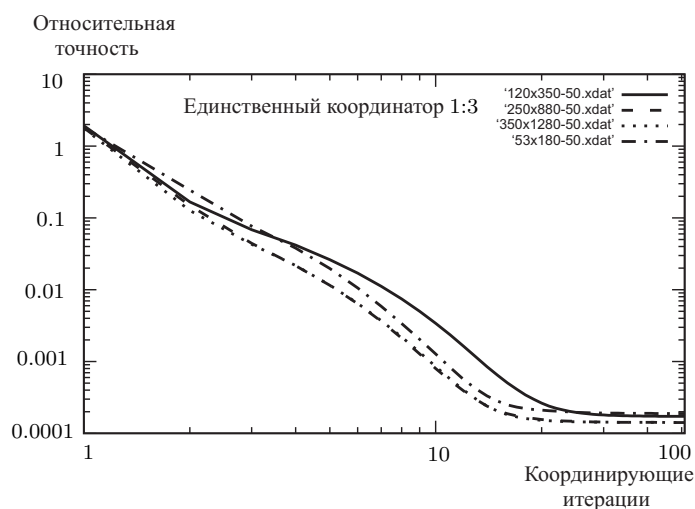


Рис. 2. Проекция на несколько случайных конусов различной размерности с соотношением “размерность : количество образующих” порядка 1:3.

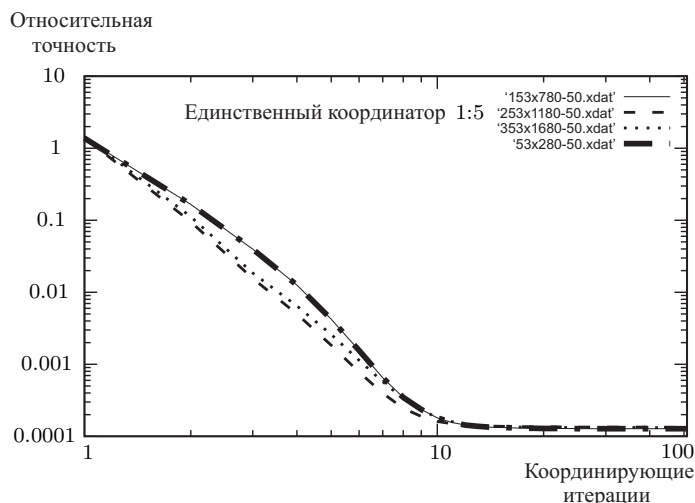


Рис. 3. Проекция на несколько случайных конусов различной размерности с соотношением “размерность : количество образующих” порядка 1:5.

3.2.2. Сходимость проекционного алгоритма с двумя координирующими векторами. На рис. 4, 5 показаны результаты тестов, проведенных с описанной программой. В качестве тестовых задач рассматривались проекции на случайные конусы с достаточно разнообразными геометрическими характеристиками образующих с различным соотношением размерности пространства и количества образующих, примерно 1:3 и 1:5.

С целью экономии места абсолютные значения размерностей задач указаны непосредственно на графиках. Например, метка “350x1280-50.xdat” означает тестовую задачу размерностью 350 переменных и 1280 образующих исходного конуса, что составляет примерно 3.6 Мб.

В качестве критерия оптимальности полученного решения приведена норма разности векторов, которые получены от проекций заданной точки p на отдельные конусы, образованные примерно одним и тем же количеством образующих, дополненных координирующим вектором. Как показано в теореме 4, при работе данного метода эти проекции сходятся к общей точке, которая является решением задачи проекции на полный конус. Таким образом, сходимость нормы разности этих векторов является естественным индикатором скорости сходимости.

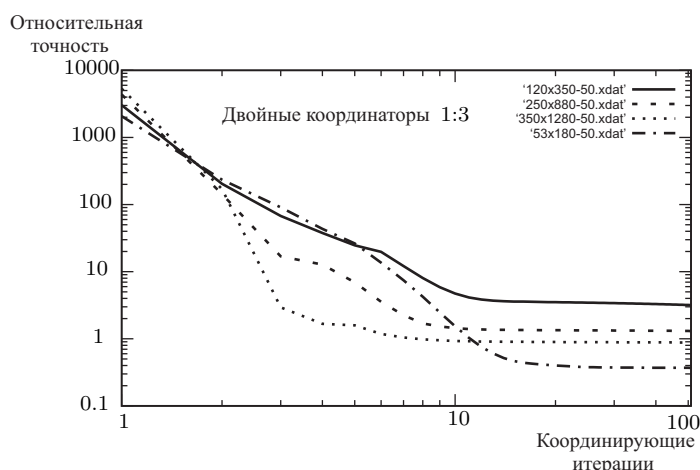


Рис. 4. Проекция на несколько случайных конусов различной размерности с соотношением “размерность : количество образующих” порядка 1:3.

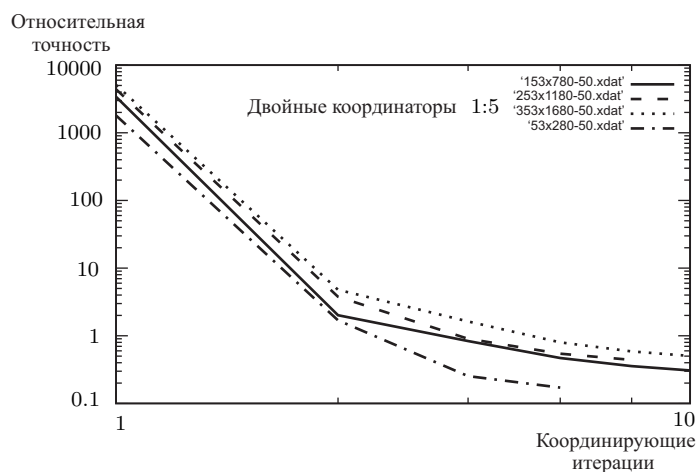


Рис. 5. Проекция на несколько случайных конусов различной размерности с соотношением “размерность : количество образующих” порядка 1:5.

При анализе вычислительных экспериментов с задачами достаточно высокой размерности представляется целесообразным соотнести разность векторов z_A^k, z_B^k с длиной собственно проектируемого вектора p , которая, конечно, растет с ростом размерности задачи. Поэтому в качестве параметра точности выбрана относительная величина $\|z_A^k - z_B^k\|/\|p\|$.

Интересным результатом этих экспериментов явились замечательная устойчивость параметров сходимости как по размерности пространства, так и по соотношениям между размерностью и количеством образующих. По этим данным скорость метода на первых 10 итерациях для задач с соотношением “размерность : количество образующих” порядка 1:3 можно оценить как устойчивую оценку порядка $O(k^{-3})$, что, конечно, достаточно неплохо.

Вместе с тем после 10-й итерации наблюдается драматическое замедление сходимости, что требует дальнейшего анализа.

Как это ни удивительно, для задач с соотношением “размерность : количество образующих” порядка 1:5 сходимость существенно быстрее (порядка $O(k^{-4})$), что можно объяснить бóльшим выбором образующих, являющихся кандидатом на ввод в базис.

Заклучение

В целом можно утверждать, что в результате проделанной работы заложена определенная теоретическая основа для создания и совершенствования алгоритмов бикомпозиции конусной проекции с возможностью существенной параллелизации вычислительных процессов. Хотя теоретической оценки вычислительной сложности приведенных алгоритмов получить пока не удалось, вычислительные эксперименты с задачами достаточно представительных размеров показывают неплохую степенную сходимости алгоритмов в диапазоне порядков 4–6. Среди проблемных вопросов с этими алгоритмами можно отметить, однако, существенное замедление сходимости при достижении средней относительной точности порядка 10^{-5} , что можно объяснить накоплением численного шума при использованной достаточно прямолинейной реализации операций линейной алгебры (факторизация Холецкого и ее модификации), отсутствием вспомогательных операций масштабирования, рефакторизации и прочих “продвинутых” приемов современной вычислительной алгебры. В свое оправдание можем заметить, что реализация и описание этих деталей вычислительного процесса, по сути дела, требуют отдельного рассмотрения и выводят работу за пределы разумных размеров журнальной статьи.

Благодарности. В заключение хотелось бы также отметить, что работа над статьей была в значительной степени инициирована в ходе выполнения программы дополнительного профессионального образования “Современные методы теории информации и оптимизации” (университет “Сириус”, Сочи, октябрь 21 — ноябрь 3, 2022). Мы весьма признательны участникам программы и университету “Сириус” за стимулирующие обсуждения и созданные условия.

СПИСОК ЛИТЕРАТУРЫ

1. NEOS Server: State-of-the-Art Solvers for Numerical Optimization. URL: <https://neos-server.org/neos/>.
2. IBM ILOG CPLEX Optimizer. URL: <https://www.ibm.com/products/ilog-cplex-optimization-studio/cplex-optimizer>.
3. The Leader in Decision Intelligence Technology – Gurobi Optimization. URL: <https://www.gurobi.com>.
4. Cardinal Optimizer (COPT): [website]. URL: <https://www.shanshu.ai/copt>.
5. GNU Linear Programming Kit (GLPK): [website]. URL: <https://www.gnu.org/software/glpk/>.
6. Open Source Linear and Mixed-Integer Programming Software and Solvers [e-resource]. URL: <https://www.gurobi.com/resources/open-source-linear-and-mixed-integer-programming-software-and-solvers/>.
7. **Nurminski E.A.** Single-projection procedure for linear optimization // J. Global Optim. 2016. Vol. 66, no. 1. P. 95–110. doi: 10.1007/s10898-015-0337-9
8. **Нурминский Е.А.** Проекция на внешне заданные полиэдры// Журн. вычисл. математики и мат. физики. 2008. Т. 48, № 3. С. 387–396.
9. **Шишкин Е.В.** Линейные пространства и отображения. М.: URSS, 2022. 312 с. ISBN: 978-5-9519-2967-9.
10. **Nurminski E.A.** Follow-up on conversion of outer projection to inner [e-resource]. doi: 10.13140/RG.2.2.28487.88489
11. **Nesterov Yu.E.** Efficiency of coordinate descent methods on huge-scale optimization problems // SIAM J. Optim. 2022. Vol. 22, no. 2. P. 341–362. doi: 10.1137/100802001
12. **Bauschke H.H., Borwein J.M.** On the convergence of von Neumann’s alternating projection algorithm for two sets // Set-Valued Anal. 1993. Vol. 1. 1993. P. 185–212.
13. **Luo Z.Q., Tseng P.** On the convergence of the coordinate descent method for convex differentiable minimization // J Optim Theory Appl. 1992. Vol. 72. P. 7–35.
14. **Wright S.J.** Coordinate descent algorithms // Mathematical Programming. 2015. Vol. 151, no. 1. P. 3–34. doi: 10.1007/s10107-015-0892-3

Поступила 25.05.2023

После доработки 8.07.2023

Принята к публикации 17.07.2023

директор программы развития Дальневосточного центра математических исследований
Дальневосточный федеральный университет
г. Владивосток
e-mail: nurminskiy.ea@dvfu.ru

REFERENCES

1. NEOS Server: State-of-the-Art Solvers for Numerical Optimization. URL: <https://neos-server.org/neos/>.
2. IBM ILOG CPLEX Optimizer. URL: <https://www.ibm.com/products/ilog-cplex-optimization-studio/cplex-optimizer>.
3. The Leader in Decision Intelligence Technology – Gurobi Optimization. URL: <https://www.gurobi.com>.
4. Cardinal Optimizer (COPT): [website]. URL: <https://www.shanshu.ai/copt>.
5. GNU Linear Programming Kit (GLPK): [website]. URL: <https://www.gnu.org/software/glpk/>.
6. Open Source Linear and Mixed-Integer Programming Software and Solvers [e-resource]. URL: <https://www.gurobi.com/resources/open-source-linear-and-mixed-integer-programming-software-and-solvers/>.
7. Nurminski E.A. Single-projection procedure for linear optimization. *J. Global Optim.*, 2016, vol. 66, no. 1, pp. 95–110. doi: 10.1007/s10898-015-0337-9.
8. Nurminski E.A. Projection onto polyhedra in outer representation. *Comp. Math. and Math. Phys.*, 2008, vol. 48, no. 3, pp. 367–375.
9. Shikin E.V. *Linejnye prostranstva i otobrazhenija* [Linear spaces and mappings]. Moscow: URSS, 2022, 312 p. ISBN: 978-5-9519-2967-9.
10. Nurminski E.A. Follow-up on conversion of outer projection to inner [e-resource]. doi: 10.13140/RG.2.2.28487.88489
11. Nesterov Yu.E. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM J. Optim.*, 2022, vol. 22, no. 2, pp. 341–362. doi: 10.1137/100802001
12. Bauschke H.H., Borwein J.M. On the convergence of von Neumann’s alternating projection algorithm for two sets. *Set-Valued Anal.*, 1993, vol. 1, 1993, pp. 185–212.
13. Luo Z.Q., Tseng P. On the convergence of the coordinate descent method for convex differentiable minimization. *J Optim Theory Appl.*, 1992, vol. 72, pp. 7–35.
14. Wright S.J. Coordinate descent algorithms. *Mathematical Programming*, 2015, vol. 151, no. 1, pp. 3–34. doi: 10.1007/s10107-015-0892-3

Received May 25, 2023

Revised July 8, 2023

Accepted July 17, 2023

Funding Agency: This work was carried out at the Far-East Mathematical Research Center and was supported by the Ministry of Science and Higher Education of the Russian Federation (agreement no. 075-02-2023-946 of February 16, 2023, for the realization of programs for the development of regional centers for mathematical research and education).

Evgeni Alekseevich Nurminski, Dr. Phys.-Math. Sci., Prof., Far Eastern Federal University, Vladivostok, 690922 Russia, email: nurminskiy.ea@dvfu.ru.

Cite this article as: E. A. Nurminski. A bicomposition of conical projections. *Trudy Instituta Matematiki i Mekhaniki UrO RAN*, 2023, vol. 29, no. 3, pp. 73–87.