

УДК 519.853.4

О РЕШЕНИИ СИСТЕМ КВАДРАТИЧНЫХ УРАВНЕНИЙ¹**А. С. Стрекаловский, М. В. Баркова**

В работе рассматривается классическая проблема решения системы квадратичных алгебраических уравнений. Для поиска решения применяется вариационный подход сведения к задаче оптимизации с функциями, представимыми разностью выпуклых функций (ДС функций). Задача оптимизации при этом оказывается невыпуклой и негладкой. Используя известные свойства ДС функций, удается свести основную задачу оптимизации к гладкой задаче ДС минимизации с ограничениями-неравенствами. Для решения последней вначале применяется специальный метод локального поиска (СМЛП), для которого исследована сходимость и обоснованы критерии останова. Тестирование СМЛП на системах с квадратичными данными продемонстрировало достаточную эффективность СМЛП, в том числе по сравнению с известными специальными пакетами прикладных программ (ППП). Процедуры глобального поиска строятся на основе условий глобальной оптимальности (УГО), позволяющих “выскакивать” из локальных решений, стационарных и критических точек. Далее произведено успешное тестирование метода глобального поиска (МГП). При этом сравнительная эффективность разработанного подхода доказана успешным решением всех тестовых систем квадратичных уравнений для задач большой размерности (с количеством уравнений и переменных до 1000). В то же время стандартные ППП на задачах большой размерности продемонстрировали свою неспособность отыскать решение в большинстве тестовых примеров.

Ключевые слова: системы квадратичных уравнений, ДС функции, условия глобальной оптимальности, локальный поиск, глобальный поиск, квадратичные задачи.

A. S. Strekalovsky, M. V. Barkova. On the solution of systems of quadratic equations.

The paper addresses the classical problem of solving a system of quadratic algebraic equations. To find a solution, we apply a variational approach of reducing the original problem to an optimization problem with cost functions represented by the difference of convex functions (DC functions). In this case, the optimization problem turns out to be nonconvex and nonsmooth. Using the known properties of DC functions, we reduce the main optimization problem to a smooth DC minimization problem with inequality constraints. To solve the latter problem, first, a special local search method (SLSM) is applied. The convergence of the method is proved and stopping criteria are established. The testing of the SLSM on systems of equations with quadratic data shows its sufficient efficiency, even when compared with known special applied software packages. Global search procedures are built on the basis of global optimality conditions, which allow us to “jump out” of local solutions and stationary and critical points. The global search method is tested. The comparative efficiency of the developed approach is proved by the successful solution of all test systems of quadratic equations for problems of large dimension (with the number of equations and variables up to 1000). At the same time, the standard applied software packages fail to solve large-dimensional problems for the most part of test examples.

Keywords: system of quadratic equations, DC functions, global optimality conditions, local search, global search, quadratic problems.

MSC: 90C26, 90C30**DOI:** 10.21538/0134-4889-2024-30-2-173-187**Введение**

Сегодня научное сообщество и цивилизация в целом находятся под впечатлением колоссального технического прогресса в компьютерных технологиях. Вместе с тем приходит все большее понимание того, что только на суперкомпьютерах решить все прикладные задачи невозможно — нужны новые вычислительные методы. Однако нетрудно видеть, что в теории численных методов подобных компьютерных “революций” и “прорывов” на рубеже XX–XXI вв. не произошло.

¹Работа выполнена в рамках базового проекта фундаментальных исследований Минобрнауки РФ (№ 121041300065-9, код FWEW-2021-0003).

Среди прикладников, например, градиентные методы оптимизации О. Коши (для решения одного полиномиального уравнения, 1847 г.) и методы Ньютона — Рафсона — Симпсона (для систем нелинейных уравнений (СНАУ), 1664, 1690, 1740 гг.) [1; 2] весьма популярны до сих пор, несмотря на их известные недостатки.

С другой стороны, на рубеже XX–XXI вв. обозначились новые вызовы [3]: поиск равновесий (скажем, по Нэшу) в конфликтных играх; поиск решения в иерархических задачах управления; поиск решений в динамических подобных задачах.

Все эти задачи оказываются невыпуклыми, и, к тому же, часто приводят к СНАУ большой размерности. Поэтому ясно, что сама невыпуклость вариационной задачи, как и СНАУ, могут быть рассмотрены как вызовы XXI века. Кроме того, СНАУ, как известно, имеют огромное число приложений, как то: проектирование технических систем, прогнозирование химических, биологических и физических процессов (например, расчет последствий ядерного взрыва), новые задачи так называемого “искусственного интеллекта” (известно, что обучение нейросети основано на решении большого (иногда гигантского) числа сложных уравнений, т. е. СНАУ), задачи геологического и климатического прогнозирования и т. д. (см. [4–11] и литературу в этих публикациях). В то же время хорошо известно, что современные методы выпуклой оптимизации (градиентные, Ньютона, последовательное квадратичное программирование (SQP), методы доверительной области (TRM) и т. д.) оказываются неоперабельными при поиске именно глобальных решений невыпуклых задач оптимизации и управления [4–15]. К тому же, широкую популярность у прикладников и молодых исследователей приобрели *B&B* методики и приемы фрагментации допустимой области [9; 10; 12–17], ведущие к “проклятию размерности”, когда объем вычислений растет экспоненциально с ростом размерности задачи. Кроме того, генетические алгоритмы и другие “биоиницированные” подходы заполонили известные журналы, хотя не имеют никакого математического обоснования.

В данной работе предложен один из вариантов вариационного (оптимизационного) подхода к решению систем квадратичных уравнений на основе теории глобального поиска, разработанной для задач оптимизации с DC функциями [18–21]. Особо обращаем внимание на статью [22], где впервые дается теоретическое обоснование общей схемы глобального поиска (СГП), сочетающей теорию точного штрафа [18–22] с условием “приближенной допустимости” [21–23], а также процедур глобального поиска, основанных на условиях глобальной оптимальности (УГО). Поэтому данную работу можно рассматривать как одно из первых тестирований общей СГП [22]. Результаты численного эксперимента подтверждают возможность решения систем квадратичных уравнений достаточно большой размерности ($n \leq 10^3$), в то время как другие популярные методы решения демонстрируют работоспособность только лишь на системах с $n = 3, 5, 10$ [9; 10]. В статье приводятся результаты численного тестирования метода глобального поиска (МГП) и его сравнения с известными специальными пакетами прикладных программ (ППП) STRSCNE [24] и Fsolve (Matlab). Эти результаты показывают значительное сравнительное преимущество разработанного подхода, что позволяет надеяться на эффективность метода для общих систем нелинейных алгебраических уравнений.

1. Постановка и DC декомпозиция задачи

Рассмотрим стандартную систему нелинейных уравнений [4; 18]

$$f_i(x) = 0, \quad i \in \mathcal{I} = \{1, 2, \dots, m\}, \quad (1.1)$$

где $x \in \mathbb{R}^n$, а функции $f_i(\cdot)$ непрерывны на некотором выпуклом открытом множестве $\Omega \subset \mathbb{R}^n$ (где допустимо, что $\Omega = \mathbb{R}^n$) и, кроме того, представимы в виде разности $f_i(x) = g_i(x) - h_i(x)$, где $g_i(\cdot)$, $h_i(\cdot)$ — выпуклые на Ω функции, $i \in \mathcal{I}$. Применим вариационный подход к решению

системы уравнений (1.1) с редукцией к задаче минимизации

$$(\mathcal{P}) \quad F(x) := \sum_{i=1}^m |f_i(x)| = \sum_{i=1}^m |g_i(x) - h_i(x)| \downarrow \min, \quad x \in \Omega. \quad (1.2)$$

При этом функция $F : \mathbb{R}^n \rightarrow \mathbb{R}$ также оказывается DC функцией: $F(x) = G(x) - H(x)$, $x \in \Omega$, а задача $(\mathcal{P})$ –(1.2), соответственно, задачей DC минимизации [16–18; 20]. Действительно, используя очевидные равенства

$$|g_i(x) - h_i(x)| \triangleq \max\{g_i(x) - h_i(x); \quad h_i(x) - g_i(x)\} = 2 \max\{g_i(x), h_i(x)\} - [g_i(x) + h_i(x)],$$

нетрудно получить следующую DC декомпозицию целевой функции задачи $(\mathcal{P})$ –(1.2):

$$F(x) = G(x) - H(x) = 2 \sum_{i=1}^m \max\{g_i(x), h_i(x)\} - \sum_{i=1}^m [g_i(x) + h_i(x)] \downarrow \min_x, \quad x \in \Omega, \quad (1.2')$$

где выпуклые [25] функции $G(\cdot)$ и $H(\cdot)$ определяются формулами

$$G(x) := 2 \sum_{i=1}^m \max\{g_i(x); \quad h_i(x)\}, \quad H(x) := \sum_{i=1}^m (g_i(x) + h_i(x)). \quad (1.3)$$

Поэтому для решения задачи $(\mathcal{P})$ –(1.2), эквивалентной решению системы (1.1), ниже применяется теория глобального поиска [18; 20; 21; 23].

Системы квадратичных уравнений

Рассмотрим систему уравнений с квадратичными невыпуклыми функциями следующего вида:

$$f_i(x) = \langle x, C_i x \rangle + \langle d^i, x \rangle + \rho_i = 0, \quad i \in \mathcal{I} = \{1, 2, \dots, m\}, \quad (1.4)$$

где $n \times n$ -матрицы C_i , $i \in \mathcal{I}$, являются, вообще говоря, знаконеопределенными, так что функции $f_i(\cdot)$, $i \in \mathcal{I}$, оказываются невыпуклыми. Существуют различные процедуры декомпозиции знаконеопределенной матрицы C_i в виде разности двух положительно определенных матриц $C_i = A_i - B_i$, $A_i, B_i > 0 \quad \forall i \in \mathcal{I}$ и как следствие DC разложения функции $f_i(\cdot) = g_i(\cdot) - h_i(\cdot)$, $i \in \mathcal{I}$ (см. например, [18], где матрицы A_i и B_i строятся положительно определенными с положительными элементами и диагональным доминированием). Здесь будем использовать следующие составляющие $g_i(\cdot)$ и $h_i(\cdot)$ DC разложения функций $f_i(x)$:

$$g_i(x) = \langle x, A_i x \rangle + \frac{\langle d^i, x \rangle}{2} + \frac{\rho_i}{2}, \quad h_i(x) = \langle x, B_i x \rangle - \frac{\langle d^i, x \rangle}{2} - \frac{\rho_i}{2}, \quad i \in \mathcal{I}. \quad (1.5)$$

Такие DC декомпозиции оказываются удобными при численном решении, поскольку функция $H(\cdot)$ из (1.3), аккумулирующая все невыпуклости задачи $(\mathcal{P})$ –(1.2'), не содержит линейных членов и констант, что облегчает организацию вычислительного процесса. Действительно, обозначив $Q := \sum_{i=1}^m (A_i + B_i)$, получим в (1.3) функции $G(\cdot)$, $H(\cdot)$ вида

$$G(x) = 2 \sum_{i=1}^m \max \left\{ \langle x, A_i x \rangle + \frac{\langle d^i, x \rangle}{2} + \frac{\rho_i}{2}; \quad \langle x, B_i x \rangle - \frac{\langle d^i, x \rangle}{2} - \frac{\rho_i}{2} \right\}, \quad (1.6)$$

$$H(x) = \langle x, Q x \rangle.$$

2. Специальный метод локального поиска

Задача $(\mathcal{P})$ –(1.2) в силу невыпуклости может обладать большим количеством локальных минимумов и стационарных точек [4; 12; 17; 25]. В таких задачах классические методы выпуклой оптимизации оказываются неоперабельными и неэффективными в смысле отыскания глобального решения задачи $(\mathcal{P})$ –(1.2) и продуцируют (в зависимости от стартовой точки) приближенно стационарные векторы (если $\|\nabla F(x^k)\| \leq \varepsilon$), допустимые точки (в задачах с ограничениями) или вообще недопустимые векторы (см. [12–17]). Поэтому численное решение, например, задач DC минимизации требует новых подходов и методов.

Рассмотрим известный метод локального поиска [18; 20; 21; 23] для задачи $(\mathcal{P})$ –(1.2), использующий последовательное решение линейаризованных по базовой невыпуклости задач следующего вида:

$$(\mathcal{PL}_s) \quad \Phi_s(x) = G(x) - \langle \nabla H(x^s), x \rangle \downarrow \min, \quad x \in \mathbb{R}^n. \quad (2.1)$$

При этом линейаризованная задача $(\mathcal{PL}_s)$ –(2.1) оказывается выпуклой и может быть решена с помощью современных методов выпуклой оптимизации [4; 12; 25], а также пакетов прикладных программ (например, Gurobi, Matlab, CPLEX и т. д.). Итак, если известна итерация $x^s \in \mathbb{R}^n$, то следующая x^{s+1} , $s = 0, 1, 2, \dots$, строится по правилу

$$\begin{aligned} \Phi_s(x^{s+1}) &\triangleq G(x^{s+1}) - \langle \nabla H(x^s), x^{s+1} \rangle \\ &\leq \inf_x \{G(x) - \langle \nabla H(x^s), x \rangle \mid x \in \mathbb{R}^n\} + \delta_s \triangleq \mathcal{V}(\mathcal{PL}_s) + \delta_s, \end{aligned} \quad (2.2)$$

где $\mathcal{V}(\mathcal{PL}_s) := \mathcal{V}_s = \inf_x \{G(x) - \langle \nabla H(x^s), x \rangle \mid x \in \mathbb{R}^n\}$ — оптимальное значение задачи $(\mathcal{PL}_s)$, а числовая последовательность $\{\delta_s\}$ (δ_s — точность решения задачи $(\mathcal{PL}_s)$) удовлетворяет условиям $\delta_s > 0$, $s = 0, 1, 2, \dots$, $\sum_{i=0}^{\infty} \delta_s < +\infty$. При этом из выпуклости функции $H(\cdot)$ и неравенства (2.2) вытекает следующая (основная) цепочка:

$$\begin{aligned} -\delta_s &\leq \mathcal{V}(\mathcal{PL}_s) - \Phi_s(x^{s+1}) \\ &\leq \Phi_s(x^s) - \Phi_s(x^{s+1}) \triangleq G(x^s) - G(x^{s+1}) - \langle \nabla H(x^s), x^s - x^{s+1} \rangle \\ &\leq G(x^s) - G(x^{s+1}) + H(x^{s+1}) - H(x^s) = F(x^s) - F(x^{s+1}), \end{aligned} \quad (2.3)$$

откуда следует, что $F(x^{s+1}) \leq F(x^s) + \delta_s$, $s = 0, 1, 2, \dots$ ($\delta_s \downarrow 0$). Это означает, что числовая последовательность $\{F(x^s)\}$ оказывается “почти” ($\delta_s > 0$) монотонно убывающей и в силу леммы 2.3.2 из [4] обладает конечным пределом $\lim_{s \uparrow \infty} F(x^s)$, поскольку функция $F(\cdot)$ ограничена снизу нулем ($F(x) \geq 0$). При этом цепочка (2.3) подсказывает выбор критерия останова в виде

$$\Phi_s(x^s) - \Phi_s(x^{s+1}) \leq \frac{\tau}{2}, \quad (2.4)$$

так как в (2.2) на каждом шаге метода вычисляются значения целевых функций $\Phi_s(x^s)$ и $\Phi_s(x^{s+1})$ задачи $(\mathcal{PL}_s)$. Далее, если неравенство (2.4) выполнено и $\delta_s \leq \frac{\tau}{2}$, то из цепочки (2.3) следует, что

$$0 \leq \mathcal{V}(\mathcal{PL}_s) - \Phi_s(x^{s+1}) \triangleq \inf_x \{G(x) - G(x^{s+1}) - \langle \nabla H(x^s), x - x^{s+1} \rangle \mid x \in \mathbb{R}^n\} \leq \frac{\tau}{2} + \delta_s \leq \tau,$$

поэтому вектор x^{s+1} оказывается τ -решением линейаризованной задачи $(\mathcal{PL}_s)$, что вполне удовлетворительно при малом $\tau > 0$. В этом случае будем называть вектор x^{s+1} τ -критической точкой метода локального поиска (2.2).

Как известно, DC декомпозицию функции $f(x) = g(x) - h(x)$ всегда можно изменить таким образом, чтобы “антывыпуклая” компонента $h(\cdot)$ была сильно выпуклой. Поэтому, если хотя

бы для одного $i = i_0 \in \mathcal{I}$ $g_i(\cdot)$ или $h_i(\cdot)$ сильно выпукла, то функция $H(x) \triangleq \sum_{i \in \mathcal{I}} [g_i(x) + h_i(x)]$ оказывается также сильно выпуклой [4; 25]. В этом случае с помощью цепочки (2.3) из неравенства сильной выпуклости для $H(\cdot)$ [4] следует, что $\lim_{s \uparrow \infty} x^s = x_*$, где $x_* \in \mathbb{R}^n$ [18; 20; 21].

Более того, в этом случае x_* оказывается решением линеаризованной (в предельной точке x_*) задачи [20; 23]

$$(\mathcal{PL}_*) \quad \Phi_*(x) = G(x) - \langle \nabla H(x_*), x \rangle \downarrow \min_x, \quad x \in \mathbb{R}^n.$$

Далее назовем такой вектор $x_* \in \mathbb{R}^n$ *критической точкой метода локального поиска* (2.2). В этом случае справедливы следующие предельные соотношения [18; 20; 21; 23]:

$$\begin{aligned} \lim_{s \uparrow \infty} F(x^s) &=: F_* \geq \mathcal{V}(\mathcal{P}) \geq 0; \\ \lim_{s \uparrow \infty} \mathcal{V}_s &= \mathcal{V}_* \geq V(\mathcal{P}); \\ \lim_{s \uparrow \infty} \Delta \Phi_s &= 0, \quad \text{где } \Delta \Phi_s = \Phi_s(x^s) - \Phi_s(x^{s+1}). \end{aligned}$$

Последние условия дают дополнительное обоснование критерию останова (2.4). Кроме того, необходимо принять во внимание следующие два факта [20; 21; 23, с. 25]. Во-первых, из (1.3) следует, что функция $G(x)$ недифференцируема. Однако задача $(\mathcal{PL}_s)$ –(2.1), как нетрудно видеть, оказывается равносильной выпуклой задаче оптимизации [19–21; 23]

$$(\mathcal{APL}(x^s)) \quad \left. \begin{aligned} \Psi_s(x, \gamma) &= 2 \sum_{i=1}^m \gamma_i - \langle \nabla H(x^s), x \rangle \downarrow \min_{(x, \gamma)}, \\ g_i(x) &\leq \gamma_i, \quad h_i(x) \leq \gamma_i, \quad i \in \mathcal{I}, \\ x &\in \mathbb{R}^n, \quad \gamma = (\gamma_1, \gamma_2, \dots, \gamma_m) \in \mathbb{R}^m, \end{aligned} \right\} \quad (2.5)$$

где решение (x_{s+1}, γ_{s+1}) задачи $(\mathcal{APL}_s)$ удовлетворяет условию

$$\gamma_i^{s+1} = \max \{g_i(x_{s+1}), h_i(x_{s+1})\}, \quad i \in \mathcal{I}.$$

Подчеркнем, что задача $(\mathcal{APL}_s)$ –(2.5) выпукла и при этом выполнено усиление условия Слейтера, т. е. $\forall \hat{x} \in \mathbb{R}^n \exists \hat{\gamma} = (\hat{\gamma}_1, \dots, \hat{\gamma}_m) \in \mathbb{R}^m: g_i(\hat{x}) < \hat{\gamma}_i, h_i(\hat{x}) < \hat{\gamma}_i, i \in \mathcal{I}$. Поэтому численное решение задачи $(\mathcal{APL}_s)$ –(2.5) может быть исполнено классическими и современными методами выпуклой оптимизации [4; 12] и пакетами прикладных программ.

Кроме того, используя квадратичное представление (1.4)–(1.6), гладкую задачу $(\mathcal{APL}(y))$ можно переписать следующим образом:

$$\left. \begin{aligned} 2 \left[\sum_{i=1}^m \gamma_i - \langle yQ, x \rangle \right] &\downarrow \min_{(x, \gamma)}, \quad (x, \gamma) \in \mathbb{R}^{n+m}, \quad y \in \mathbb{R}^n, \\ \langle x, A_i x \rangle + \frac{\langle d^i, x \rangle}{2} + \frac{\rho_i}{2} &\leq \gamma_i, \quad \langle x, B_i x \rangle - \frac{\langle d^i, x \rangle}{2} - \frac{\rho_i}{2} \leq \gamma_i, \quad i \in \mathcal{I}, \end{aligned} \right\} \quad (2.6)$$

так что на каждом шаге СМЛП необходимо решать гладкую выпуклую задачу (2.6) с переменными $(x, \gamma) \in \mathbb{R}^{n+m}$ и с $2m$ выпуклыми квадратичными ограничениями-неравенствами, которая решается стандартными методами, например, квадратичной оптимизации [4].

3. Численное тестирование метода локального поиска

Ниже представлены результаты тестирования специального метода локального поиска. Для вычислительного эксперимента были взяты 5 примеров из набора тестовых систем квадратичных уравнений из [26], решения которых заранее известны. Все системы уравнений имеют

вид (1.1) со следующими функциями $f_i(\cdot)$:

- 1) $f_i(x) = (3 - 2x_i)x_i + 1 - x_{i-1} - 2x_{i+1}$, $x_0 = x_{n+1} = 0$, $i = 1, \dots, n$;
- 2) $f_i(x) = 3x_i - x_{i-1} - 2x_{i+1} - \kappa x_i^2 + 1$, $x_0 = x_{n+1} = 0$;
- 3) $f_1(x) = 1 - x_1$, $f_i(x) = 10(i-1)(x_i - x_{i-1})^2$, $i = 2, 3, \dots, n$;
- 4) $f_i(x) = 1 - x_i$, $i = 1, 3, 5, \dots, n-1$,
 $f_i(x) = 10(x_i - x_{i-1}^2)$, $i = 2, 4, \dots, n$, где n четное;
- 5) $f_i(x) = 3x_i(x_{i+1} - 2x_i + x_{i-1}) + \frac{(x_{i+1} - x_{i-1})^2}{4}$, $x_0 = 0$; $x_{n+1} = 20$, $i = 1, 2, \dots, n$.

Для каждой из тестовых систем была сформулирована оптимизационная задача, а также в явном виде представлено DC разложение квадратичных функций $f_i(\cdot)$, $i \in \mathcal{I}$ (согласно разд. 1). Обозначим через $(\mathcal{P}_l)$, $l = 1, 2, 3, 4, 5$, задачу $(\mathcal{P})$ –(1.2'), соответствующую системе квадратичных уравнений с функциями $f_i(\cdot)$, $i \in \mathcal{I}$ вида 1)–5).

СМЛП был протестирован на задачах $(\mathcal{P}_l)$, $l = 1, 2, 3, 4, 5$, где использовались следующие стартовые точки:

$$\begin{aligned}
 x_0^{(1)} &= (1, 1, \dots, 1)^\top; \\
 x_0^{(2)} &= (0, 0, \dots, 0)^\top; \\
 x_0^{(3)} &= (1, -1, 1, \dots, (-1)^{n+1})^\top; \\
 x_0^{(4)} &= (10, 10, \dots, 10)^\top; \\
 x_0^{(5)} &= (-1.2, 1, -1.2, 1, \dots, 1)^\top; \\
 x_0^{(6)} &= (1, -2, 3, -4, \dots, (-1)^{n+1}n)^\top; \\
 x_0^{(7)} &= \left(1, 3, 5, \dots, 2\left\lfloor \frac{n+1}{2} \right\rfloor - 1, 2, 4, \dots, 2\left\lfloor \frac{n+1}{2} \right\rfloor\right)^\top, \text{ где } \lfloor \cdot \rfloor \text{ — целая часть числа.}
 \end{aligned}$$

Для решения вспомогательных выпуклых задач (2.6) (с точностью $\delta_s = 10^{-5}$) использовался пакет программ Gurobi 9.5.1. Специальный метод локального поиска реализован на языке C++, все вычисления проводились на ПК с процессором Intel Core i5-4670K 3.4GHz.

Подробно с результатами тестирования СМЛП для размерностей $n = 10, 50, 100, 500, 800, 1000$ можно ознакомиться по ссылке [27], где для каждой задачи $(\mathcal{P}_i)$, $i = 1, 2, 3, 4, 5$, представлены значения целевой функции в стартовой и в полученной τ -критической точке. Значительное, а для некоторых точек колоссальное, улучшение значения функции в приближенно критической точке позволяет сделать выводы об эффективности разработанного подхода уже на этапе локального поиска.

В таблицах также представлены результаты работы современных ППП для решения систем нелинейных уравнений. Это встроенный Matlab-решатель для CHAY Fsolve, использующий метод Левенберга — Марквардта, а также солвер STRSCNE [24], использующий комбинацию метода Ньютона — Рафсона с методом доверительной области. Оба ППП реализованы в среде Matlab.

Результаты вычислительного эксперимента [27] показали, что СМЛП нашел решение систем с функциями вида 1), 2) (т.е. задач $(\mathcal{P}_1)$ и $(\mathcal{P}_2)$), стартуя из некоторых начальных приближений. Однако для задачи $(\mathcal{P}_3)$ СМЛП ни в одном случае не удалось найти решение системы уравнений. Тестовый пример с функциями вида 4) оказался тривиальным (СМЛП нашел решение системы с уравнениями вида 4), начиная работу из большинства стартовых точек). Думается, такие результаты обусловлены структурой систем с функциями 4), где в отличие от остальных тестовых примеров количество квадратичных уравнений оказывается вдвое меньше, чем количество переменных ($m = n/2$). Поэтому далее тестирование данной системы не будет рассматриваться. В то же время специализированные ППП (Fsolve, STRSCNE) не смогли отыскать решение системы нелинейных уравнений с функциями 4), начиная поиск из некоторых точек на размерности $n \geq 100$. Стартуя из начальных приближений $x_0^{(6)}$ и $x_0^{(7)}$, солвер Fsolve для систем всех размерностей $n \geq 100$ останавливался, не доходя до решения.

Схожую ситуацию можно проследить и у солвера STRSCNE, так как начиная с размерности $n > 100$ он не смог отыскать приближенное решение системы нелинейных уравнений, останавливаясь в точке, значение целевой функции в которой близко к 10^{-2} . Непривлекательной для дальнейших испытаний оказалась и система с функциями вида 5), поскольку в большинстве случаев и СМЛП, и ППП нашли ее решение (несмотря на то, что в соответствующей ей оптимизационной задаче ($\mathcal{P}_5$) матрица C имеет трехдиагональную структуру). Поэтому задача ($\mathcal{P}_5$) была исключена из дальнейшего тестирования.

Ниже представлены сводные таблицы сравнения работы СМЛП с результатами, полученными с помощью ППП Fsolve и солвера STRSCNE, где используются следующие обозначения: $\#$ — номер стартовой точки $x_0^{(\#)}$, N_{svd} — количество решенных СМЛП задач ($\mathcal{P}_l$), $l = 1, 2, 3, 4, 5$ (т. е. значение целевой функции задачи равно нулю с точностью $\varepsilon = 10^{-3}$); в случае ППП N_{svd} — количество решенных ППП систем квадратичных уравнений вида 1)–5) (т. е. $0 \leq N_{svd} \leq 5$); T_A — среднее время работы алгоритмов для одной задачи в секундах.

Т а б л и ц а 1

Результаты тестирования СМЛП и ППП на задачах невысокой размерности

$n = 10$						$n = 50$						$n = 100$						
CM/JH			Fsolve		STRSCNE	CM/JH			Fsolve		STRSCNE	CM/JH			Fsolve		STRSCNE	
#	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A
1	2	0.11	4	0.01	4	0.02	1	0.31	2	0.03	3	0.57	1	0.67	2	0.09	3	1.85
2	4	0.01	5	0.01	5	0	4	0.04	4	0.08	5	0.02	3	0.1	4	0.30	5	0.10
3	3	0.02	3	0.43	4	0.11	2	0.05	3	0.05	4	0.07	2	0.11	2	0.14	4	0.20
4	1	0.05	4	0.02	4	0.02	2	0.1	3	0.09	3	0.61	1	0.49	3	0.32	3	1.29
5	2	0.03	4	0.02	4	0.01	2	0.07	2	0.07	4	0.06	2	0.16	2	0.19	4	0.25
6	2	0.06	3	0.02	4	0.01	1	0.05	1	0.06	1	0.64	2	0.12	0	0.16	1	1.57
7	1	0.2	4	0.02	4	0.02	2	0.05	2	0.07	2	0.55	2	0.13	1	0.17	2	1.28

Т а б л и ц а 2

Результаты тестирования СМЛП и ППП на задачах повышенной размерности

$n = 500$						$n = 800$						$n = 1000$								
CM/JH			Fsolve		STRSCNE		CM/JH			Fsolve		STRSCNE		CM/JH			Fsolve		STRSCNE	
#	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A
1	2	21.7	2	1.1	3	34.4	2	96.9	2	3.2	3	76.7	2	300.4	2	5.9	3	116.2		
2	3	6.2	2	6.7	5	110.2	3	18.0	2	20.3	5	237.9	3	874.5	2	43.9	4	368.9		
3	2	3.5	1	2.1	3	3.6	2	12.7	1	6.1	3	9.6	2	541.8	1	10.3	4	3.9		
4	1	18.2	1	7.7	3	23.2	1	72.9	1	22.8	3	118.7	2	702.3	2	39.9	1	65.0		
5	2	7.4	2	2.9	3	3.6	2	28.6	2	7.9	3	14.9	2	747.5	2	12.6	3	15.7		
6	2	6.9	0	1.9	0	21.5	3	20.3	0	5.9	0	107.4	3	816.9	0	11.7	1	183.7		
7	2	6.3	0	2.3	0	21.4	3	20.1	0	5.2	0	120.8	3	818.0	0	10.1	0	45.2		

Т а б л и ц а 3

Уменьшение целевой функции на этапе локального поиска

#	$n = 10$		$n = 50$		$n = 100$		$n = 500$		$n = 800$		$n = 1000$	
	LP_A	ΔF_A	LP_A	ΔF_A	LP_A	ΔF_A	LP_A	ΔF_A	LP_A	ΔF_A	LP_A	ΔF_A
1	40	32.94	40	43.04	39	67.04	39	219.04	40	333.04	3	409.04
2	8	25.01	9	45.01	10	69.19	14	269.19	10	419.19	12	517.07
3	9	427.48	7	10138.06	7	40278.07	7	$1.0 \cdot 10^6$	7	$2.5 \cdot 10^6$	8	$4.0 \cdot 10^6$
4	48	1344.52	42	6738.08	46	13319.75	41	66268.39	43	$1.1 \cdot 10^5$	9	$1.3 \cdot 10^5$
5	19	526.04	16	12200.54	16	48573.36	16	$1.2 \cdot 10^6$	16	$3.1 \cdot 10^6$	10	$4.8 \cdot 10^6$
6	55	20155.19	7	$1.2 \cdot 10^7$	12	$1.9 \cdot 10^8$	15	$1.25 \cdot 10^{11}$	11	$8.2 \cdot 10^{11}$	14	$1.9 \cdot 10^{12}$
7	97	1421.01	11	$1.8 \cdot 10^5$	12	$1.4 \cdot 10^6$	13	$1.8 \cdot 10^8$	11	$7.53 \cdot 10^8$	14	$1.5 \cdot 10^9$

Из табл. 1 и 2 можно видеть, что среднее время работы СМЛП для задач небольшой размерности составляет доли секунды, постепенно возрастаая с увеличением размерности. В то же время среднее число решенных задач для каждой стартовой точки остается приблизительно одинаковым вне зависимости от размерности. Стоит отметить, что уже на этапе локального поиска с помощью предложенного подхода СМЛП удалось найти решение как минимум в 20% задач, начиная поиск из любого начального приближения. Чего нельзя сказать о пакетах прикладных программ Fsolve и STRSCNE, которые демонстрируют эффективность только для задач небольшой размерности и для определенных стартовых точек. Так, начиная с размерности $n > 100$ (см. табл. 2) при старте с точек $x_0^{(6)}$ и $x_0^{(7)}$, ППП не смогли отыскать решение систем квадратичных уравнений (исключение составляет одна задача — система уравнений типа 4)) размерности $n = 1000$, решенная солвером STRSCNE). В то же время эти стартовые точки ($x_0^{(6)}$ и $x_0^{(7)}$), из которых локальный поиск и ППП не достигли решения, представляют особенный интерес при тестировании глобального поиска, поскольку создают наихудшие условия для работы алгоритма. Тем самым полученные результаты еще раз подчеркивают зависимость эффективности работы современных решателей от выбора близкого к корню стартового приближения.

В табл. 3 представлены средние величины разности целевой функции в стартовой и полученной с помощью СМЛП точках для каждой размерности, где LP_A — среднее число решенных линеаризованных задач; $\Delta F_A := (F(x_0) - F(z))/35$ — усредненная разность значений целевой функции в стартовой и полученной точках. В некоторых случаях можно отметить гигантский “скачок” разности значений уже на этапе локального поиска, в то время как среднее число решенных линеаризованных задач не превышает 97.

Таким образом, по результатам вычислительного эксперимента можно сделать выводы об эффективности представленного выше СМЛП при решении задач высокой размерности, которому в некоторых случаях уже при поиске критической точки $x(\tau)$ (локальным поиском) в задаче (P) удалось найти глобальное решение.

4. Глобальный поиск

Одна из главных проблем современной оптимизации состоит в создании новых технологий и методов, позволяющих “выскакивать” из локальных ям, в которые попадают классические методы [9; 13–17]. Используемый здесь новый подход, основанный на условиях глобальной оптимальности (УГО) [18; 19; 21; 22], заключается в разработке: а) новых методов локального поиска, “внутри” которых работают классические и современные методы выпуклой оптимизации (градиентные, Ньютона, последовательное квадратичное программирование (SQP), методы доверительной области (TRM) и т. д.), а также используется линеаризация по базовой невыпуклости исследуемой задачи; б) вычислительных процедур выхода из “локальной ямы”, основанных на УГО.

Основой численных процедур выхода из критических точек является тот факт [19, теорема 2], что для вектора $z \in \mathbb{R}^n$, не являющегося ε -решением задачи (P) –(1.2): $z \notin \varepsilon\text{-Sol}(P)$, т. е. $\zeta := F(z) > \varepsilon > 0 = \mathcal{V}(P)$, при условии существования $v \in \mathbb{R}^n : F(v) > F(z) - \varepsilon = \zeta - \varepsilon$, непременно найдутся параметры $(y, \beta) \in \mathbb{R}^{n+1}$ и вектор $u \in \mathbb{R}^n$, для которых

$$(a) H(y) = \beta - \zeta + \varepsilon; \quad (b) G(y) \leq \beta; \quad (c) G(u) - \beta < \langle \nabla H(y), u - y \rangle, \quad (4.1)$$

т. е. нарушается основное неравенство из УГО, что позволяет выскочить из “локальной ямы” с помощью конструктивного свойства УГО.

Кроме того, нетрудно видеть, что вышеупомянутые условия сводят решение невыпуклой задачи (P) –(1.2) к решению семейства линеаризованных выпуклых задач

$$(PL(y)) \quad \Phi_y(x) := G(x) - \langle \nabla H(y), x \rangle \downarrow \min_x, \quad x \in \mathbb{R}^n. \quad (4.2)$$

(где параметры “возмущения” (y, β) удовлетворяют условиям (4.1) (a), (b)), а также последующей проверке основного неравенства УГО (точнее его нарушения (4.1) (c)).

Все вышеуказанные свойства обосновывают следующие этапы глобального поиска [22].

4.1. Схема глобального поиска

Пусть заданы стартовая точка $x^0 \in \mathbb{R}^n$ и числовые последовательности $\{\tau_k\}, \{\delta_k\} > 0$, $k = 1, 2, \dots$, $\{\tau_k\} \downarrow 0$, $\{\delta_k\} \downarrow 0$ ($k \rightarrow \infty$), а также точность решения $\varepsilon > 0$.

0. Положить $k := 0$, $x^k = x^0 \in \mathbb{R}^n$.

I. Стартуя из $x^k \in \mathbb{R}^n$, $k := 0$, с помощью специального метода локального поиска найти τ_k -критический вектор $z^k \in \mathbb{R}^n$.

II. Выбрать число $\beta \in [\beta_-, \beta_+]$, где $\beta_- = F(z^k) = \zeta_k$, $\beta_+ := \sup(G, \Omega)$ ($\Omega \subset \mathbb{R}^n$) может быть оценено достаточно грубо для $k = 0$, а затем уточняться на каждой итерации.

III. Построить конечную аппроксимацию

$$\mathcal{A}_k(\beta) := \{v^1, \dots, v^N \mid H(v^i) = \beta - \zeta_k, i = 1, \dots, N_k, N_k(\beta)\} \quad (4.3)$$

поверхности уровня функции $H(\cdot)$ ($H(v) = \beta - \zeta_k$) и сформировать множество индексов

$$I_k := I_k(\beta) := \{i \in \{1, 2, \dots, N_k\} : G(v^i) \leq \beta\}.$$

IV. Для каждого $i \in I_k$ найти приближенное δ_k -решение $\bar{u}^i$ линейризованной (выпуклой) задачи $(PL(v^i))$ –(4.2) ($\bar{u}^i \in \delta_k\text{-Sol}(PL(v^i))$)

$$(PL(v^i)) \quad G(x) - \langle \nabla H(v^i), x \rangle \downarrow \min_x, \quad x \in \mathbb{R}^n. \quad (4.4)$$

V. Для каждого $i \in I_k$, стартуя из точки $\bar{u}^i$, с помощью СМЛП найти $2\tau_k$ -критическую точку $u^i \in \mathbb{R}^n$.

VI. Для каждого $i \in I_k$ найти приближенное δ_k -решение w^i задачи уровня (с целью проверки неравенства (4.1)(c))

$$(Lev\mathcal{P}_k) \quad \langle \nabla H(v), u^i - v \rangle \uparrow \max_v, \quad H(v) = \beta - \zeta_k, \quad i = 1, \dots, N_k.$$

VII. Проверить $\forall i = 1, \dots, N_k$ основное неравенство условий оптимальности [18; 19; 21]

$$G(u^i) - \beta \geq \langle \nabla H(w^i), u^i - w^i \rangle. \quad (4.5)$$

VIII. Если нашелся номер $j \in \{1, \dots, N_k\}$, при котором неравенство (4.5) нарушено, то положить $k := k + 1$, $x^{k+1} := u^j$ и вернуться на шаг I. В случае отсутствия такого j изменить значение $\beta \in [\beta_-, \beta_+]$ (скажем, одномерным поиском) и вернуться на шаг III. Если неравенство (4.5) справедливо $\forall \beta \in [\beta_-, \beta_+]$ (т.е. метод одномерного поиска по β финишировал), то положить $x^{k+1} := z^k$, $k := k + 1$ и вернуться на шаг 1.

Отметим, что при работе с квадратичными функциями задача уровня, возникающая на шаге VI, может быть решена аналитически. Рассмотрим особенности реализации схемы глобального поиска (см. [18; 20; 21; 23]).

4.2. Аппроксимации поверхности уровня функции $H(\cdot)$

На шаге III схемы глобального поиска необходимо построить аппроксимацию (4.3) поверхности уровня функции $H(\cdot)$, а именно набор точек v^j таких, что

$$H(v^j) = \langle v^j, Qv^j \rangle = \beta - \zeta_k, \quad j = 1, 2, \dots, N_k, \quad \beta > \zeta_k. \quad (4.6)$$

С учетом опыта ранее решенных задач [18; 21] использовались следующие аппроксимации.

Первая аппроксимация использует единичные орты $e^j = (0, \dots, 0, \overset{j}{1}, 0, \dots, 0)^\top$, $j = 1, \dots, n$, задающие базис в пространстве $\mathbb{R}^n$:

$$\mathcal{A}_1 = \{(v^1, \dots, v^{2n}) \mid v^j = -\mu_i e^j, v^{j+n} = \mu_i e^j, H(v^l) = \beta - \zeta_k, l = \overline{1, 2n}\}. \quad (4.7)$$

Нетрудно заметить, что при увеличении размерности n такая зависимость приводит к увеличению объема вычислений, так как для каждой точки аппроксимации на этапе IV (пока не нашлась пара (y, β) и вектор u , удовлетворяющие (4.1)) необходимо решать выпуклую квадратичную задачу оптимизации ($PL(y)$), а после запускать метод локального поиска.

Построение второй аппроксимации основано на информации о текущей критической точке z^k , а также на множестве векторов $\{p_j, j = 1, 2, \dots, n\}$, являющихся попарно сопряженными относительно матрицы Q : $\langle p^j, Qp^l \rangle = 0 \quad \forall j \neq l$, так что

$$\mathcal{A}_2 = \{(v^1, \dots, v^{2n}) \mid v^j = z^k - \mu_j p^j, v^{j+n} = z^k + \mu_j p^j, H(v^l) = \beta - \zeta_k, l = \overline{1, 2n}\}. \quad (4.8)$$

Для аппроксимаций $\mathcal{A}_1$ и $\mathcal{A}_2$ параметр μ_j , вычисляется таким образом, чтобы было выполнено условие (4.6). Поскольку построение попарно сопряженных векторов p^j , $j = 1, 2, \dots, n$, не зависит от текущей точки z^k , то его можно выполнить всего один раз в начале вычислений.

Отметим, что построение аппроксимации поверхности уровня функции $H(\cdot)$ не исчерпывается вышеуказанными правилами построения в исследуемой задаче. Эффективность алгоритма глобального поиска (АГП) с использованием той или иной аппроксимации рассматривается ниже при представлении результатов вычислительного эксперимента.

5. Численное тестирование алгоритма глобального поиска

В данном разделе представлены результаты вычислительного эксперимента по тестированию метода глобального поиска в задаче $(\mathcal{P})$ –(1.2). На *первом этапе* эксперимента были решены задачи $(\mathcal{P}_i)$, $i = 1, 2, 3$, размерности от $n = 10$ до 100. Цель этого этапа — проверить работоспособность предлагаемого подхода, а также подобрать наиболее эффективную (из предложенных выше (4.7) и (4.8)) аппроксимацию поверхности уровня.

При этом были выбраны следующие параметры алгоритма. Точность работы локального поиска τ_k была выбрана 10^{-4} , точность глобального поиска — $\varepsilon = 10^{-3}$ (см. в [18; 21] согласование точностей $\varepsilon_k \geq 2\delta_k, \tau_k \geq 2\delta_k$). Одномерный поиск по β осуществлялся на отрезке $[\zeta_k, n \times 10^4]$ с шагом $\Delta\beta = (100 \times n)/2$. На этом этапе вычислительного эксперимента тестирование проводилось с использованием тех стартовых точек, из которых на этапе локального поиска не было найдено глобальное решение задачи $(\mathcal{P})$ –(1.2), поэтому число тестируемых задач для различных случаев отличается.

Из результатов вычислительного эксперимента по сравнению работы АГП с применением аппроксимаций $\mathcal{A}_1$ и $\mathcal{A}_2$ для каждой из задач $(\mathcal{P}_i)$, $i = 1, 2, 3, 4, 5$ (см. [27]), можно видеть, что на размерности $n = m = 10$ и $n = m = 50$ для задачи $(\mathcal{P}_1)$ с использованием наборов точек $\mathcal{A}_1$ и $\mathcal{A}_2$ АГП удалось найти глобальное решение задачи. Однако уже на размерности $n = 80$ АГП с аппроксимацией $\mathcal{A}_1$ удается выйти из критической точки и найти глобальное решение, стартуя только из начального приближения $x_0^{(1)}$. При дальнейшем увеличении размерности, АГП, использующий аппроксимацию $\mathcal{A}_1$, не находит глобального решения ни в одном из случаев. Таким образом, для задачи $(\mathcal{P}_1)$ АГП с аппроксимацией $\mathcal{A}_2$ выигрывает. Стартуя из всех начальных точек, он находит глобальное решение вне зависимости от размерности, чего нельзя сказать об использовании более простой аппроксимации $\mathcal{A}_1$. Для задачи $(\mathcal{P}_2)$ во всех случаях АГП удалось найти глобальное решение с использованием как аппроксимации $\mathcal{A}_1$, так и $\mathcal{A}_2$. Результаты тестирования задачи $(\mathcal{P}_3)$ подчеркивают преимущество выбора аппроксимации $\mathcal{A}_2$, использующей сопряженные относительно матрицы Q векторы. Действительно, АГП, использующий вторую аппроксимацию, нашел решение системы квадратичных уравнений 3) во всех

Т а б л и ц а 4

Сравнение эффективности аппроксимации

n	N	$\mathcal{A}_1$					$\mathcal{A}_2$				
		N_{svd}	Itr_A	Loc_A	PL_A	T_A	N_{svd}	Itr_A	Loc_A	PL_A	T_A
10	15	9	29	1132	42583	102.34	15	3	15	81	0.09
50	17	11	6	6538	711589	5690.06	17	3	7	29	0.16
80	18	6	3	21978	1305176	16401.20	18	3	7	32	0.31
100	18	5	2	28954	1677609	29303.34	18	3	7	32	0.37

Т а б л и ц а 5

Результаты тестирования АГП и ППП Fsolve и STRSCNE

n	$F_A(x_0)$	N	АГП		Fsolve		STRSCNE	
			N_{svd}	T_A	N_{svd}	T_A	N_{svd}	T_A
10	519.29	15	15	0.09	8	0.02	9	0.02
50	26039.51	17	17	0.16	4	0.09	6	0.68
80	731577.14	18	18	0.31	2	0.25	4	1.23
100	196368.37	18	18	0.37	2	0.30	4	1.72
500	23896142.09	18	18	15.39	0	5.63	4	59.56
800	97699858.09	16	16	56.23	0	17.28	4	153.43
1000	190761173.51	16	16	3798.48	0	31.87	4	182.86

случаях. В то время как АГП с аппроксимацией $\mathcal{A}_1$ не смог отыскать решение ни в одном из примеров.

Ниже представлены сводные результаты сравнения работы АГП с применением аппроксимаций $\mathcal{A}_1$ и $\mathcal{A}_2$. В табл. 4 используются следующие обозначения: N — количество задач участвовавших в тестировании (для которых на этапе локального поиска не было найдено решение); N_{svd} — количество решенных задач; Itr_A — среднее число итераций глобального поиска (число улучшений целевой функции); Loc_A — среднее количество запусков СМЛП (число пройденных критических точек); PL_A — среднее число решенных линеаризованных задач; T_A — среднее время работы алгоритма в секундах для одной задачи.

Результаты из табл. 4 наглядно демонстрируют преимущество аппроксимации $\mathcal{A}_2$ по всем параметрам. АГП, использующему эту аппроксимацию, удалось найти решение для всех примеров в то время как использование аппроксимации $\mathcal{A}_1$ позволяет решить меньше половины задач. Более того, вне зависимости от размерности ему потребовалось не более 3 итераций глобального поиска, чтобы найти решение во всех случаях (стартуя из всех начальных приближений).

Таким образом, аппроксимация $\mathcal{A}_2$, построенная на основе сопряженных относительно матрицы Q векторов, выглядит более надежной, находя решение систем уравнений для всех случаев любой размерности. Поэтому для второго этапа тестирования глобального поиска было принято решение использовать эту аппроксимацию.

Второй этап эксперимента посвящен решению тестовых систем уравнений с квадратичными функциями вида **1)–3)** с использованием аппроксимации $\mathcal{A}_2$. По результатам из сводной табл. 5 можно видеть, что АГП удалось решить все предложенные задачи с указанной точностью до размерности $n = 1000$ за приемлемое время, в то время как специализированные ППП не смогли решить больше половины тестовых систем. В столбце $F_A(x_0)$, указано среднее значение целевой функции в стартовой точке для задачи каждой размерности, которое показывает удаленность начального значения от нуля. Такой “скачок” целевой функции из стартовой точки в решение показывает, что за небольшое число итераций глобальному поиску удастся найти решение, начиная поиск, например для задач размерности $n = 1000$, со значения $1.9 \cdot 10^8$! Это еще раз подтверждает сравнительную эффективность применения разработанной методики для решения систем квадратичных уравнений повышенной размерности.

Заключение

Выше был представлен вариант нового вариационного подхода к классической проблеме численного решения системы нелинейных алгебраических уравнений, имеющих широкое поле известных приложений. Оригинальность исходной постановки задачи заключается в том, что каждая функция $f_i(\cdot)$, $i \in \mathcal{I}$, представима в виде разности выпуклых функций, а функция невязки $F(x)$ представляется суммой модулей, что необычно для стандартной методики решения подобных систем ввиду негладкости $F(x)$. Тем не менее удалось показать, что целевая функция также является DC функцией (т. е. разностью выпуклых функций).

Далее был предложен новый (упрощенный, см. [20; 23]) специальный метод локального поиска (СМЛП) для этого типа задач, и исследована его сходимость к критической точке с соответствующим критерием останова. Кроме того, проведено численное тестирование СМЛП на известной серии тестовых задач, продемонстрировавшее его сравнительную эффективность по отношению к известным решателям (ППП) Fsolve и STRSCNE на задачах размерности $n \leq 10^3$.

СНАУ рассматривается как частный случай общей задачи из [22]. Разработан оригинальный подход к решению СНАУ, базирующийся на условиях глобальной оптимальности (УГО) [18; 19; 22] и состоящий из двух этапов: (а) локального поиска; (б) процедур выхода из критических точек (полученных СМЛП).

Проведен вычислительный эксперимент (результаты которого представлены в [27]), продемонстрировавший значительную сравнительную эффективность разработанной методики решения СНАУ на квадратичных тестовых задачах размерности $n \leq 10^3$. В частности, новый алгоритм глобального поиска решил все тестовые задачи, в то время как решатели даже на небольших размерностях $n = 10, 50, 80, 100$ не смогли решить все квадратичные задачи, стартуя из семи начальных приближений.

Представляется, что эти результаты “открывают” двери для решения СНАУ не только с квадратичными, но и с полиномиальными и любыми непрерывными функциями достаточно высокой размерности (скажем, $n = 10^4$ – 10^5). Тестирование новой методики будет продолжено и в этом направлении.

СПИСОК ЛИТЕРАТУРЫ

1. **Lemarechal C.** Cauchy and the gradient method // Documenta Math. 21st Int. Symp. Math. Program. Berlin, August 19–24. 2012. Extra Vol. Optimization Stories. P. 251–254.
2. **Deuffhard P.** A short history of newton’s method // Documenta Math. 21st Int. Symp. Math. Program. Berlin, August 19–24. 2012. Extra Vol. Optimization Stories. P. 25–30.
3. **Pang J.-S.** Three modeling paradigms in mathematical programming // Math. Program. 2010. Vol. 124, no. 2. P. 297–323.
4. **Васильев Ф.П.** Методы оптимизации. М.: МЦНМО, 2011. 620 с.
5. **Бахвалов Н.С., Н.П. Жидков, Г.М. Кобельков** Численные методы. М.: Бином, 2013. 640 с.
6. **Тыртышников Е.Е.** Методы численного анализа. М.: Академия, 2007. 320 с.
7. **Самарский А.А., Гулин А.В.** Численные методы. М.: Наука, 1989. 432 с.
8. **Ортега Дж., Рейнбольд В.** Итерационные методы решения нелинейных систем уравнений. М.: Мир, 1975. 559 с.
9. **Floudas C., Klepeis J., Pardalos P.** Global optimization approaches in protein folding and peptide docking // DIMACS Ser. in Discrete Math. and Theoret. Comp. Sci. 1999. Vol. 47. P. 141–171.
10. **Pei J., Drazic Z., Drazic M., Mladenovic N., Pardalos P.** Continuous variable neighborhood search (C-VNS) for solving systems of nonlinear equations // INFORMS J. Comp. 2019. Vol. 31, no. 2. P. 235–250.
11. **Канторович Л.В., Акилов Г.П.** Функциональный анализ. М.: Наука; Издание 2-е, перераб., 1977. 744 с.
12. **Nocedal J., Wright St.J.** Numerical optimization. NY: Springer, 2006. 664 p.
13. **Byrd R.H., Marazzi M., Nocedal J.** On the convergence of Newton iterations to non-stationary points // Math. Program., Ser. A. 2004. Vol. 99, no. 1. P. 127–148. doi: 10.1007/s10107-003-0376-8

14. **Mascarenhas W. F.** The BFGS method with exact line searches fails for non-convex objective functions // *Math. Program., Ser. A.* 2004. Vol. 99, no. 1. P. 49–61. doi: 10.1007/s10107-003-0421-7
15. **Wachter A., Biegler L.T.** Failure of global convergence for a class of interior point methods for nonlinear programming // *Math. Program., Ser. B.* 2000. Vol. 88, no. 3. P. 565–574. doi: 10.1007/PL00011386
16. **Floudas C.A., Pardalos P.M. et al.** Handbook of test problems in local and global optimization. NY: Springer Science and Business Media Dordrecht, 1999. 442 p. doi: 10.1007/978-1-4757-3040-1
17. **Horst R., Tuy H.** Global optimization: Deterministic approaches. Berlin: Springer-Verlag, 1996. 730 p. doi: 10.1007/978-3-662-03199-5
18. **Стрекаловский А.С.** Элементы невыпуклой оптимизации. Новосибирск: Наука, 2003. 356 с.
19. **Стрекаловский А.С.** Новые условия глобальной оптимальности задачи с d.c. ограничениями // *Тр. Ин-та математики и механики УрО РАН.* 2019. Т. 25, № 1. С. 245–261. doi: 10.21538/0134-4889-2019-25-1-245-261
20. **Strekalovsky A.S.** Local search for nonsmooth DC optimization with DC equality and inequality constraints // *Constraints Numerical Nonsmooth Optimization: State of the Art Algorithms* / eds. A.M. Bagirov, M. Gaudioso, N. Karimtsa et al. Cham: Springer Internat. Publ., 2020. P. 229–261. doi: 10.1007/978-3-030-34910-3_7
21. **Strekalovsky A.S.** On solving optimization problems with hidden nonconvex structures // *Optimization in Science and Engineering* / eds. T.M. Rassias, C.A. Floudas, S. Butenko. NY: Springer, 2014. P. 465–502. doi: 10.1007/978-1-4939-0808-0_23
22. **Стрекаловский А.С.** Минимизирующие последовательности в задаче DC оптимизации с ограничениями // *Тр. Ин-та математики и механики УрО РАН.* 2023. Т. 29, № 3. С. 185–209. doi: 10.21538/0134-4889-2023-29-3-185-209
23. **Strekalovsky A.S., Minarchenko I.M.** A local search method for optimization problem with d.c. inequality constraints // *Appl. Math. Modelling.* 2018. Vol. 58. P. 229–244. doi: 10.1016/j.apm.2017.07.031
24. **Bellavia S., Macconi M., Morini B.** STRSCNE: A scaled trust region solver for constrained nonlinear equations // *Comput. Optim. Appl.* 2004. Vol. 28, no. 1. P. 31–50.
25. **Hiriart-Urruty J.-B., Lemarechal C.** Convex analysis and minimization algorithms. Berlin: Springer-Verlag, 1993. 418 p. doi: 10.1007/978-3-662-02796-7
26. **Roose A., Kulla V., Lomp M., Meressoo T.** Test examples of systems of non-linear equations. Tallin: Estonian Software and Computer Service Company, 1990.
27. Дополнительные материалы к статье: Стрекаловский А.С., Баркова М.В. “О решении систем квадратичных уравнений”. Онлайн-версия содержит дополнительные материалы, размещенные на http://journal.imm.uran.ru/Suppl_inf_2024-v.30-2 (PDF) .

Поступила 2.10.2023

После доработки 13.03.2024

Принята к публикации 18.03.2024

Стрекаловский Александр Сергеевич

д-р физ.-мат. наук, профессор

зав. отделом

Институт динамики систем и теории управления имени В.М. Матросова СО РАН

г. Иркутск

e-mail: strek@icc.ru

Баркова Мария Владимировна

младший науч. сотрудник

Институт динамики систем и теории управления имени В.М. Матросова СО РАН

г. Иркутск

e-mail: mbarkova@icc.ru

REFERENCES

1. Lemarechal C. Cauchy and the gradient method. *Documenta Math.*, 21st Int. Symp. Math. Program., Berlin, 2012, Extra Vol. Optimization Stories, pp. 251–254.

2. Deuffhard P. A short history of Newton's method. *Documenta Math. 21st Int. Symp. Math. Program.*, Berlin, 2012, Extra Vol. Optimization Stories, pp. 25–30.
3. Pang J.-S. Three modeling paradigms in mathematical programming. *Math. Program., Ser. B*, 2010, vol. 125, no. 2, pp. 297–323. doi: 10.1007/s10107-010-0395-1
4. Vasil'ev F.P. *Metody optimizatsii* [Optimization methods]. Moscow, MCNMO, 2011, 620 p. ISBN 978-5-94057-706-5.
5. Bahvalov N.S., Zhidkov N.P., Kobelkov G.M. *Chislennyye metody* [Numerical Methods]. Moscow, Binom Publ., 2013, 640 p.
6. Tyrtysnikov E.E. *Metody chislennogo analiza* [Methods of numerical analysis]. Moscow, Academia Publ., 2007, 317 p. ISBN: 978-5-7695-3925-1.
7. Samarskii A.A., Gulin A.V. *Chislennyye metody* [Numerical Methods]. Moscow, Nauka Publ., 1989, 429 p. ISBN: 5-02-013996-3.
8. Ortega J.M., Rheinboldt W.C. *Iterative solution of nonlinear equations in several variables*. NY, Acad. Press, 1970, 572 p. Translated to Russian under the title *Iteratsionnyye metody resheniya nelineynykh sistem uravneniy*, Moscow, Mir Publ., 1975, 559 p.
9. Floudas C., Klepeis J., Pardalos P. Global optimization approaches in protein folding and peptide docking. *DIMACS Ser. in Discrete Math. and Theoret. Comp. Sci.*, 1999, vol. 47, pp. 141–171.
10. Pei J., Drazic Z., Drazic M., Mladenovic N., Pardalos P. Continuous variable neighborhood search (C-VNS) for solving systems of nonlinear equations. *INFORMS J. Comput.*, 2019, vol. 31, no. 2, pp. 235–250.
11. Kantorovich L.V., Akilov G.P. *Functional Analysis*. NY, Pergamon Press, 1982, 604 p. doi: 10.1016/C2013-0-03044-7. Original Russian text published in Kantorovich L.V., Akilov G.P. *Funktsional'nyi analiz*, Moscow, Nauka Publ., 1977, 744 p.
12. Nocedal J., Wright St.J. *Numerical optimization*. NY, Springer, 2006, 664 p. doi: 10.1007/978-0-387-40065-5
13. Byrd R.H., Marazzi M., Nocedal J. On the convergence of Newton iterations to non-stationary points. *Math. Program., Ser. A.*, 2004, vol. 99, no. 1, pp. 127–148. doi: 10.1007/s10107-003-0376-8
14. Mascarenhas W.F. The BFGS method with exact line searches fails for non-convex objective functions. *Math. Program., Ser. A.*, 2004, vol. 99, no. 1, pp. 49–61. doi: 10.1007/s10107-003-0421-7
15. Wachter A., Biegler L.T. Failure of global convergence for a class of interior point methods for nonlinear programming. *Math. Program., Ser. B.*, 2000, vol. 88, no. 3, pp. 565–574. doi: 10.1007/PL00011386
16. Floudas C.A., Pardalos P.M. et al. *Handbook of test problems in local and global optimization*. NY, Springer Science and Business Media Dordrecht, 1999, 442 p. doi: 10.1007/978-1-4757-3040-1
17. Horst R., Tuy H. *Global optimization: deterministic approaches*. Berlin, Springer-Verlag, 1996, 730 p. doi: 10.1007/978-3-662-03199-5
18. Strekalovskii A.S. *Elementy nevy pukloi optimizatsii* [Elements of nonconvex optimization]. Novosibirsk, Nauka Publ., 2003, 356 p. ISBN: 5-02-032064-1.
19. Strekalovsky A.S. New global optimality conditions in a problem with d.c. constraints. *Tr. Instituta Matematiki i Mekhaniki UrO RAN*, 2019, vol. 25, no. 1, pp. 245–261 (in Russian). doi: 10.21538/0134-4889-2019-25-1-245-261
20. Strekalovsky A.S., Bagirov A.M. et al. Local search for nonsmooth DC optimization with DC equality and inequality constraints. In: *Constraints Numerical Nonsmooth Optimization: State of the Art Algorithms*, eds. A. M. Bagirov, M. Gaudioso, N. Karmita et al. Cham: Springer Internat. Publ., 2020. P. 229–261. doi: 10.1007/978-3-030-34910-3_7
21. Strekalovsky A.S., Rassias T.M. et al. On solving optimization problems with Hidden nonconvex structures. In: *Optimization in Science and Engineering*. NY, Springer, 2014, pp. 465–502. doi: 10.1007/978-1-4939-0808-0_23
22. Strekalovsky A.S. Minimizing sequences in a constrained DC optimization problem. *Proc. Steklov Inst. Math. (Suppl.)*, 2023, vol. 323, suppl. 1, pp. S255–S278. doi: 10.1134/S0081543823060214
23. Strekalovsky A.S., Minarchenko I.M. A local search method for optimization problem with d.c. inequality constraints. *Appl. Math. Model.*, 2018, vol. 58, pp. 229–244. doi: 10.1016/j.apm.2017.07.031
24. Bellavia S., Macconi M., Morini B. STRSCNE: A scaled trust region solver for constrained nonlinear equations. *Comput. Optimiz. Appl.*, 2004, vol. 28, no. 1, pp. 31–50.
25. Hiriart-Urruty J.-B., Lemaréchal C. *Convex analysis and minimization algorithms. I*, Berlin, Springer-Verlag, 1993, 418 p. doi: 10.1007/978-3-662-02796-7
26. Roose A., Kulla V., Lomp M., Meressoo T. *Test examples of systems of non-linear equations*. Tallin, Estonian Software and Computer Service Company, 1990. ISBN: 9789916031643.

27. Supplementary Materials for the article Strekalovsky A.S., Barkova M.V. “On the solution of systems of quadratic equations”. The online version contains supplementary material available at http://journal.imm.uran.ru/Suppl_inf_2024-v.30-2 (PDF).

Received October 2, 2023

Revised March 13, 2024

Accepted March 18, 2024

Funding Agency: The research was carried out within a basic project for fundamental research of the Ministry of Science and Higher Education of the Russian Federation (no. 121041300065-9, code FWEW-2021-0003).

Alexander Sergeevich Strekalovsky, Dr. Phys.-Math. Sci., Prof., Matrosov Institute for System Dynamics and Control Theory of the Siberian Branch of the Russian Academy of Sciences, Irkutsk, 664033 Russia, e-mail: strek@icc.ru.

Maria Vladimirovna Barkova, Matrosov Institute for System Dynamics and Control Theory of the Siberian Branch of the Russian Academy of Sciences, Irkutsk, 664033 Russia, e-mail: mbarkova@icc.ru.

Cite this article as: A.S.Strekalovsky, M.V.Barkova. On the solution of systems of quadratic equations. *Trudy Instituta Matematiki i Mekhaniki UrO RAN*, 2024, vol. 30, no. 2, pp. 173–187.