

ОПТИМИЗАЦИОННЫЙ ПОДХОД К РЕШЕНИЮ СИСТЕМ НЕЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ

М. В. Янулевич, А. С. Стрекаловский

Институт динамики систем и теории управления имени В.М. Матросова СО РАН, 664033, Иркутск

УДК 519.615.5+519.853.4

Рассматривается задача численного решения системы нелинейных алгебраических уравнений. Исследуемая задача редуцируется к оптимизационной задаче, для решения которой применяется теория глобального поиска, основанная на необходимых и достаточных условиях глобальной оптимальности для задач д.с. минимизации. Метод глобального поиска включает в себя два основных этапа: локальный поиск и процедуру улучшения (критических) точек, полученных локальным поиском. Проведен вычислительный эксперимент, в ходе которого исследована эффективность работы методов локального и глобального поисков на различных тестовых задачах отыскания решения систем нелинейных уравнений.

Ключевые слова: системы нелинейных алгебраических уравнений, невыпуклая оптимизация, д.с. функции, локальный поиск, глобальный поиск.

Введение

Несмотря на широкий спектр методов [1–3], разработанных для решения систем нелинейных алгебраических уравнений, проблема численного поиска решений остается весьма актуальной. Например, при применении методов типа Ньютона [1, 4–6], возникает трудность в выборе подходящего начального приближения, обеспечивающего сходимость к решению. Причем с ростом размерности системы сложность поиска стартовой точки многократно возрастает.

В случае использования вариационных методов мы имеем дело с экстремальной задачей, которая на практике оказывается невыпуклой. Уже на протяжении нескольких десятилетий для специалистов препятствием являются невыпуклым задачам оптимизации [7–11], в которых может существовать достаточно большое количество локальных решений и стационарных точек, весьма далеких от глобального решения по значению целевой функции. Выпуклые же задачи оптимизации, как известно, обладают свойством, что каждое локальное решение является глобальным [4].

Для невыпуклых задач оптимизации прямое применение методов выпуклой оптимизации (ньютоновских, сопряженных градиентов, барьерных и др.) может давать непредсказуемые последствия [12]. Тем не менее, существуют и другие подходы к решению невыпуклых экстремальных задач [10, 13–17]. В частности, обратим внимание на разрабатываемую в последние десятилетия теорию глобального поиска [10], ядром которой являются необходимые и достаточные условия глобальной оптимальности [10, 11].

В настоящей работе для решения систем нелинейных алгебраических уравнений осуществляется их редукция к невыпуклым задачам оптимизации, которые в свою очередь решаются с применением теории глобального поиска (ТГП) [10]. Основными этапами ТГП являются локальный поиск и процедура нелокального улучшения по значению целевой функции. Кроме того, осуществляется сравнительное численное тестирование разработанных алгоритмов с известными пакетами [18, 19], показавшее эффективность применяемого подхода на серии тестовых задач.

1 Постановка задачи

Рассмотрим следующую систему уравнений:

$$f_i(x) := g_i(x) - h_i(x) = 0, \quad i = 1, 2, \dots, m, \quad (1)$$

в которой $x \in \mathbb{R}^n$, а $x \mapsto f_i(x)$ — (d.c.) функции, представимые в виде разности двух выпуклых функций: $g_i(x), h_i(x)$ — выпуклые на \mathbb{R}^n функции, $i = 1, 2, \dots, m$.

Как известно [3], система уравнений (1) может быть сведена к задаче минимизации

$$F(x) = \Phi(f_1(x), \dots, f_m(x)) \downarrow \min, \quad x \in \mathbb{R}^n, \quad (2)$$

где в качестве функций $\Phi: \mathbb{R}^m \rightarrow \mathbb{R}$ можно использовать, например, следующие функции:

$$\Phi_1(t) = \sum_{i=1}^m |t_i|, \quad \Phi_2(t) = \sum_{i=1}^m t_i^2.$$

Учитывая невыпуклость в общем случае функций $f_i(\cdot)$ (например, см. [7]), нетрудно видеть, что задача (2) оказывается невыпуклой.

Замечание 1.1. Отметим, что для численного решения системы уравнений (1) использование функции $\Phi_1(t) = \|t\|_1$, $t \in \mathbb{R}^m$, выглядит предпочтительным. Действительно, если точка \tilde{x} является приближенным решением задачи (2) с точностью ε ($\tilde{x} \in \varepsilon\text{-Sol}(2)$):

$$F(\tilde{x}) \leq \inf_x \{F(x) \mid x \in \mathbb{R}^n\} + \varepsilon = \varepsilon,$$

то в случае использования функции $\Phi_1(t) = \|t\|_1$, $t \in \mathbb{R}^m$, мы получаем, что $|f_i(\tilde{x})| \leq \varepsilon \forall i = 1, 2, \dots, m$. Если же выбрать функцию $\Phi_2(t) = \|t\|_2^2$, $t \in \mathbb{R}^m$, то при тех же условиях мы получим, что $|f_i(\tilde{x})| \leq \varepsilon^{\frac{1}{2}} \forall i = 1, 2, \dots, m$. Другими словами, если удалось найти решение оптимизационной задачи (2), например, с точностью $\varepsilon = 10^{-4}$, что в случае выбора функции $\Phi_1(\cdot)$ каждое уравнение системы (1) выполняется с той же точностью 10^{-4} , а в случае выбора $\Phi_2(\cdot)$ каждое уравнение системы (1) выполняется с точностью 10^{-2} . \square

Далее в работе для сведения системы уравнений (1) к оптимизационной задаче (2) мы будем использовать функцию

$$\Phi(t) := \Phi_w(t) = \sum_{i=1}^m w_i |t_i|, \quad (3)$$

где $w = (w_1, w_2, \dots, w_m)^\top$ — вектор весовых коэффициентов, который вводится с целью решения проблемы масштабирования значений функции $f_i(\cdot)$ (например, см. [20]). В этой случае целевая функция задачи (2) является негладкой.

Используя известные свойства d.c. функций [7, 8, 10, 21], нетрудно видеть, что целевая функция $F(\cdot)$ задачи (2) является d.c. функцией, т.е. представима в виде разности двух выпуклых функций:

$$F(x) = G(x) - H(x), \quad (4)$$

где функции

$$G(x) = 2 \sum_{i=1}^m w_i \max\{g_i(x), h_i(x)\}, \quad H(x) = \sum_{i=1}^m w_i (g_i(x) + h_i(x)) \quad (5)$$

являются выпуклыми. Таким образом, для решения задачи

$$(\mathcal{P}): \quad F(x) = G(x) - H(x) \downarrow \min, \quad x \in \mathbb{R}^n, \quad (2')$$

являющейся эквивалентной системе уравнений (1), может быть использована теория глобального поиска для d.c. минимизации [10, 17], в соответствии с которой поиск глобального решения в задаче (\mathcal{P}) состоит из двух основных этапов: локального поиска и процедуры нелокального улучшения, основанной на условиях глобальной оптимальности.

2 Метод локального поиска

Метод локального поиска для задачи d.c. минимизации (\mathcal{P}) состоит в последовательном (приближенном) решении задач

$$(\mathcal{P}L_s): \quad \Phi_s(x) = G(x) - \langle \nabla H(x^s), x \rangle \downarrow \min_x, \quad x \in \mathbb{R}^n. \quad (6)$$

Зная точку x^s , можно построить точку x^{s+1} как приближенное решение задачи (6):

$$\Phi_s(x^{s+1}) \leq \inf_x \{\Phi_s(x) \mid x \in \mathbb{R}^n\} + \delta_s, \quad (7)$$

где числовая последовательность $\{\delta_s\}$ такая, что $\delta_s > 0$, $s = 0, 1, 2, \dots$, $\sum_{i=0}^{\infty} \delta_i < +\infty$.

Можно показать [10, 17, 22], что в случае сильной выпуклости [23] функции $H(\cdot)$ последовательность $\{x^s\}$, генерируемая методом локального поиска, сходится к точке x_* , являющейся решением линейаризованной задачи

$$\Phi_*(x) = G(x) - \langle \nabla H(x_*), x \rangle \downarrow \min_x, \quad x \in \mathbb{R}^n. \quad (8)$$

Точку x_* в этом случае будем называть критической точкой в задаче (\mathcal{P}_0) .

В качестве критерия останова метода локального поиска будем использовать следующее неравенство:

$$\Phi_s(x^s) - \Phi_s(x^{s+1}) + \delta_s \leq \tau, \quad (9)$$

которое обеспечивает то, что точка x^s является δ_s -критической, т.е. является δ_s -приближенным решением задачи $(\mathcal{P}L_s)$:

$$\Phi_s(x^s) \leq \inf_x \{\Phi_s(x) \mid x \in \mathbb{R}^n\} + \tau.$$

С учетом d.c. разложения (4)–(5) выпуклую задачу $(\mathcal{P}L_s)$ можно представить в следующем виде:

$$\Phi_s(x) = 2 \sum_{i=1}^m w_i \max\{g_i(x), h_i(x)\} - \sum_{i=1}^m w_i \langle \nabla g_i(x^s) + \nabla h_i(x^s), x \rangle \downarrow \min_x, \quad x \in \mathbb{R}^n. \quad (6')$$

Данная задача является, вообще говоря, негладкой задачей оптимизации, для решение которой можно использовать соответствующие методы негладкой оптимизации. Тем не менее, в работе предлагается осуществлять решение задачи $(\mathcal{P}L_s)$ без использования методов негладкой оптимизации, редуцируя негладкие задачи к гладким [5] и применяя известные методы и пакеты программ для выпуклой оптимизации (в частности, IBM ILOG CPLEX [24] или Gurobi [25]).

3 Системы квадратичных уравнений

С целью проведения численного тестирования метода локального поиска были рассмотрены следующие системы систем нелинейных уравнений с квадратичными функциями:

$$f_i(x) = \frac{1}{2} \langle x, C_i x \rangle + \langle d^i, x \rangle + \gamma_i = 0, \quad i = 1, 2, \dots, m, \quad (10)$$

где $(n \times n)$ -матрицы C_i , $i = \overline{1, m}$, являются, вообще говоря, знаконеопределенными. Как известно [10], их можно представить в виде разности двух положительно определенных матриц: $C_i = A_i - B_i$, $A_i, B_i > 0 \forall i \in \{1, 2, \dots, m\}$. Далее, система (10) редуцируется к оптимизационной задаче (\mathcal{P}) , где целевая функция $F(\cdot)$, которая представима в виде разности двух выпуклых функций:

$$G(x) = \sum_{i=1}^m \max\{\langle x, A_i x \rangle + \langle d^i, x \rangle + \gamma_i, \langle x, B_i x \rangle - \langle d^i, x \rangle - \gamma_i\}, \quad H(x) = \frac{1}{2} \sum_{i=1}^m \langle x, (A_i + B_i)x \rangle.$$

Нетрудно видеть, что функция $G(\cdot)$, $j = 1, 2$, является негладкой, а функция $H(\cdot)$ дифференцируема.

Для произвольного фиксированного вектора $y \in \mathbb{R}^n$ необходимо уметь решать следующую вспомогательную линейаризованную выпуклую задачу:

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

Вообще говоря, существует бесконечно много представлений матриц C_i , $i = 1, 2, \dots, m$, в виде разности двух положительно определенных матриц. В работе будем использовать разложение матриц C_i , $i = 1, 2, \dots, m$, в соответствии с процедурой, описанной, в частности, в [10].

Кроме того, разложение матриц C_i , $i = 1, 2, \dots, m$, можно осуществить, например, следующим образом: $C_i = A_i - B_i$, где $A_i = C_i + \mu_i I$, $B_i = \mu_i I$, $i = 1, 2, \dots, m$. При этом параметр $\mu_i \geq 0$ полагается равным $\mu_i := -\min\{0, \lambda_{\min}(C_i)\} + \chi$, $\chi \geq 0$, где $\lambda_{\min}(C_i)$ — минимальное собственное значение матрицы C_i , $i = \overline{1, m}$.

4 Численное тестирование метода локального поиска

В данном разделе приведем результаты тестирования метода локального поиска, заключающегося в последовательном приближенном решении выпуклых задач (PL_s) –(6), $s = 0, 1, 2, \dots$. Для решения задач (PL_s) , $s = 0, 1, 2, \dots$, использовался пакет программ IBM ILOG CPLEX 12.6.2 (64 bit) [24]. Программная реализация метода локального поиска осуществлялась на языке C++. Все вычисления проводились на ПК с процессором Intel Core i5–4670K 3.4GHz и RAM 16 Gb.

Для вычислительного эксперимента были взяты примеры из набора тестовых систем нелинейных (квадратичных) уравнений из [26], решения которых заранее известны. В частности, представим результаты тестирования на следующих тестовых задачах:

$$\begin{aligned} (\mathcal{P}_1): \quad & f_i(x) = (3 - 2x_i)x_i + 1 - x_{i-1} - 2x_{i+1} = 0, \quad i = \overline{1, n}, \quad x_0 = x_{n+1} = 0, \\ (\mathcal{P}_2): \quad & f_1(x) = 1 - x_1 = 0, \quad f_i(x) = 10(i - 1)(x_i - x_{i-1})^2 = 0, \quad i = \overline{2, n}. \end{aligned}$$

Выбирались 10 различных стартовых точек с целью создать наихудшие условия для работы тестируемого алгоритма, а также для подготовки работы метода глобального поиска.

n	$N(x^0)$	$F(x^0)$	SLSM			STRSCNE			MATLAB		
			$F(\bar{x})$	It	Time	$F(\bar{x})$	It	Time	$F(\bar{x})$	It	Time
10	1	57	1.61919	30	0.2	2.56295	136	0.11	2.56360	21	0.09
10	2	9	1.37246	85	0.28	2.56295	124	0.11	2.80064	16	0.08
10	3	114	1.47488	26	0.11	2.56295	138	0.11	2.56251	18	0.08
10	4	10	0	2	0.01	0	6	0.01	0	7	0.03
10	5	1960	1.71204	6	0.01	2.80132	98	0.09	2.80150	25	0.07
10	6	8	1.37238	54	0.19	2.56295	118	0.10	2.56353	17	0.08
10	7	60	0	3	0.01	0	6	0.01	0	5	0.06
10	8	784	1.35361	64	0.22	2.56295	145	0.12	2.56351	20	0.07
10	9	748	1.71262	42	0.17	2.80132	111	0.09	2.80215	40	0.11
10	10	3422	1.71112	7	0.03	2.80132	94	0.08	2.80220	41	0.11
30	1	177	1.59832	4	0.02	2.69624	124	0.33	2.69804	41	0.58
30	2	29	1.41425	3	0.02	2.69624	113	0.29	2.89842	21	0.40
30	3	354	1.53945	6	0.02	2.69624	128	0.49	2.69794	40	0.52
30	4	30	0	3	0.02	0	6	0.08	2.07460	31	0.26
30	5	5940	1.76974	6	0.03	2.89807	103	0.27	2.90133	50	0.60
30	6	23	1.41424	3	0.01	2.69624	101	0.55	2.69833	41	0.32
30	7	180	0	3	0.02	0	6	0.21	0	5	0.31
30	8	18924	1.53936	8	0.03	2.69624	118	0.72	2.69768	46	0.66
30	9	18848	1.76981	9	0.03	2.89796	233	1.21	2.90066	52	0.36
30	10	9193.3	1.7698	7	0.03	2.89807	105	0.51	2.90090	64	0.56

Таблица 1: Результаты сравнительного тестирования метода локального поиска на задаче (\mathcal{P}_1)

Было произведено сравнительное тестирование метода локального поиска с результатами, полученными в среде MathWorks MATLAB 7.11.0.584 R2010b 64-bit с использованием стандартного решателя “fsolve” [18], а также Scaled Trust Region Solver for Constrained Nonlinear Equations (STRSCNE) [19]. Результаты сравнительного тестирования представлены в Табл. 1. Здесь и далее в таблицах используются следующие обозначения: $N(x^0)$ — номер стартовой точки x^0 ; $F(x_0)$ и $F(\bar{x})$ — значение целевой функции задачи (\mathcal{P}) в стартовой точке x^0 и полученной точке \bar{x} соответственно; It — число итераций метода (количество решенных вспомогательных выпуклых задач на этапе локального поиска); Time — время работы (в секундах).

Жирным шрифтом выделены случаи, когда уже на этапе локального поиска было получено глобальное решение. Как нетрудно видеть из Табл. 1, метод локального поиска оказался эффективнее по полученному значению целевой функции задачи (\mathcal{P}) в сравнении с решателями “fsolve” [18] и STRSCNE [19], хотя время его работы при этом было больше.

Следует отметить, что (d.c.) разложение матриц C_i имеет значительное влияние на результаты вычислительного эксперимента, которые могут существенно отличаться. Так, например, в тестовой задаче (\mathcal{P}_2), в которой все функции $f_i(\cdot)$, $i = \overline{1, m}$, являются выпуклыми, оказалось, что в случае использования весовых коэффициентов $w_i = 1$, $i = \overline{1, m}$, и разложения матриц C_i , $i = \overline{1, m}$, основанного на их минимальном собственном значении, уже на этапе локального поиска мы получаем приближенное глобальное решение с точностью 10^{-3} (см. Табл. 2). Для задачи (\mathcal{P}_2) использование весовых коэффициентов дало лишь выигрыш по времени работы метода локального поиска, но не по значению функции $F(\cdot)$.

x^0	$n = m$	$F(x^0)$	$F_w(x^0)$	СМЛП with weights				СМЛП without weights		
				$F(\bar{x})$	$F_w(\bar{x})$	PL	Time	$F(\bar{x})$	PL	Time
1	100	198000	49.5	0.98732	0.00014	7	0.12	0.00086	20	0.39
2	100	0	0	0	0	1	0.01	0	1	0.01
3	100	792001	50.5	3.94734	0.00014	7	0.13	0.00121	19	0.39
4	100	1	1	0	0	2	0.03	0	2	0.03
5	100	9	1	0.50000	0.05555	2	0.03	0	2	0.03
6	100	12375	49.5	0.06182	0.00014	7	0.13	0.00075	17	0.34
7	100	49502	50.5	0.74647	0.25013	7	0.11	0.00075	18	0.36
8	100	993283500	49.5	10381.62488	0.00014	7	0.12	0.00104	19	0.39
9	100	4900500	49.5	1.04005	0.00014	7	0.13	0.00088	20	0.41
10	100	1980.8	50.2	0.00988	0.00014	7	0.12	0.00077	15	0.31

 Таблица 2: Результаты тестирования метода локального поиска на задаче (\mathcal{P}_2)

5 Глобальный поиск

Глобальный поиск осуществлялся в соответствии с теорией глобального поиска из [10] (см. также [17])

В Табл. 3–3 представлены результаты тестирования алгоритма глобального поиска. В дополнение к предыдущим обозначениям в Табл. 3–3 будем использовать следующие: $F(z)$ — значение целевой функции в полученной точке; It_{glob} — количество итераций алгоритма глобального поиска.

x^0	$F(x^0)$	GSA				STRSCNE			MATLAB		
		$F(z)$	It_{glob}	PL	Time	$F(z)$	It	Time	$F(z)$	It	Time
1	57	0	7	976	3.12	2.56295	136	0.11	2.56360	21	0.09
2	9	0	3	888	2.78	2.56295	124	0.11	2.80064	16	0.08
3	114	0	7	969	3.10	2.56295	138	0.11	2.56251	18	0.08
5	1960	0	10	1913	6.19	2.80132	98	0.09	2.80150	25	0.07
6	8	0	4	907	2.87	2.56295	118	0.10	2.56353	17	0.08
8	784	0	19	2587	8.45	2.56295	145	0.12	2.56351	20	0.07
9	748	0	10	2102	6.76	2.80132	111	0.09	2.80215	40	0.11
10	3422	0	9	2060	6.63	2.80132	94	0.08	2.80220	41	0.11

 Таблица 3: Результаты тестирования алгоритма глобального поиска на задаче (\mathcal{P}_1), $n = 10$

Как можно видеть из результатов вычислительного эксперимента, на небольшой размерности систем уравнений (в частности, при $n = 10$) отличие от методов STRSCNE [19] и MATLAB “fsolve” [18] для всех стартовых точек алгоритм глобального поиска позволил отыскать глобальное решение задачи (\mathcal{P}), а значит и решение системы уравнений (\mathcal{P}_1).

Отметим, что с ростом размерности задачи ситуация изменяется. Вычислительный эксперимент показал (см. Табл. 4), что при размерности $n = 20$, стартуя из 2 точек (с номерами 3 и 10), алгоритмом глобального поиска не удалось найти глобальное решение (значения целевого функционала выделены курсивным шриф-

x^0	$F(x^0)$	GSA				STRSCNE			MATLAB		
		$F(z)$	It_{glob}	PL	Time	$F(z)$	It	Time	$F(z)$	It	Time
1	117	0	7	3147	11.23	2.69240	284	1.14	2.69243	32	0.38
2	19	0	7	3453	13.06	2.69240	93	0.32	2.89438	19	0.2
3	234	<i>0.83067</i>	12	4624	18.39	2.69240	81	0.19	2.69247	32	0.15
5	3950	0	24	7415	29.13	2.89458	104	0.47	2.89584	47	0.77
6	15	0	7	3592	13.48	2.69240	80	0.22	2.69239	32	0.29
8	5754	0	18	6719	27.23	2.69240	79	0.39	2.69377	31	0.26
9	5698	0	29	8743	38.18	2.89459	118	0.23	2.89457	146	0.79
10	6494	<i>0.83067</i>	22	7261	28.34	2.89459	112	0.53	2.89578	54	0.38

Таблица 4: Результаты тестирования алгоритма глобального поиска на задаче (\mathcal{P}_1) , $n = 20$

том). Отметим при этом, что алгоритму всегда все же удавалось несколько раз осуществлять нелокальные улучшения критических точек (см. колонку It_{glob}).

Тем не менее, результаты алгоритма глобального поиска оказались значительно лучше в сравнении с результатами STRSCNE [19] и MATLAB “fsolve” [18], что свидетельствует об эффективности работы алгоритма глобального поиска. При этом нужно отметить, что модификация и усовершенствование алгоритма глобального поиска будут продолжены.

Заключение

На основе теории глобального поиска [10] разработан алгоритм решения систем нелинейных уравнений, основанных на редукции к негладкой невыпуклой задаче оптимизации. Данный подход протестирован на серии тестовых систем квадратичных уравнений из [26] и произведено сравнение с известными пакетами программ [18, 19]. Результаты вычислительно эксперимента в большинстве случаев продемонстрировали высокую эффективность разработанного подхода.

Список литературы

- [1] Бахвалов Н.С., Жидков Н.П., Кобельков Г.М. Численные методы. М.: Бином. Лаборатория знаний, 2008.
- [2] Dennis J., Schnabel R. Numerical Methods for Unconstrained Optimization and Nonlinear Equations. New York: SIAM, 1996.
- [3] Ortega J., Rheinboldt W. Iterative Solution of Nonlinear Equations in Several Variables. New York: Academic Press, 1970.
- [4] Васильев Ф.П. Методы оптимизации. М.: Факториал Пресс, 2002.
- [5] Nocedal J., Wright S. Numerical Optimization. New York: Springer, 2006.
- [6] Izmailov A.F., Solodov M. Newton-Type Methods for Optimization and Variational Problems. New York: Springer, 2014.
- [7] Hiriart-Urruty J.-B. Generalized differentiability, duality and optimization for problems dealing with difference of convex functions // Convexity and Duality in Optimization / Ed. by J. Ponstein. Vol. 256 of Lecture Notes in Economics and Mathematical Systems. Berlin: Springer-Verlag, 1985. P. 37–70.
- [8] Tuy H. Convex Analysis and Global Optimization. Dordrecht: Kluwer Academic Publishers, 1998.
- [9] Handbook of Global Optimization, Nonconvex Optimization and its Applications / Ed. by R. Horst, P.M. Pardalos. Dordrecht: Kluwer Academic Publishers, 1995.
- [10] Стрекаловский А.С. Элементы невыпуклой оптимизации. Новосибирск: Наука, 2003.

- [11] Strekalovsky A.S. Global optimality conditions in nonconvex optimization // Journal of Optimization Theory and Applications. 2017. Vol. 173, No. 3. P. 770–792.
- [12] Mascarenhas W. Newton's iterates can converge to non-stationary points // Mathematical Programming. 2008. Vol. 112, No. 2. P. 327–334.
- [13] Strongin R.G., Sergeev Y.D. Global Optimization with Non-convex Constraints. Dordrecht: Kluwer Academic Publishers, 2000.
- [14] Tuy H. On solving nonconvex optimization problems by reducing the duality gap // Journal of Global Optimization. 2005. Vol. 32, No. 3. P. 349–365.
- [15] Zilinskas A. A strong homogeneity of two global optimization based on statistical models of multimodal objective functions // Applied Mathematics and Computation. 2012. Vol. 218, No. 16. P. 8131–8136.
- [16] Floudas C. Deterministic Global Optimization: Theory, Methods and Application. Dordrecht: Kluwer Academic Publishers, 1999.
- [17] Strekalovsky A.S. On solving optimization problems with hidden nonconvex structures // Optimization in Science and Engineering / Ed. by T.M. Rassias, C.A. Floudas, S. Butenko. New York: Springer, 2014. P. 465–502.
- [18] MathWorks. MATLAB. <https://www.mathworks.com/help/optim/ug/fsolve.html>. 2017. Accessed: 31.05.2017.
- [19] Bellavia S., Macconi M., Morini B. STRSCNE: A scaled trust region solver for constrained nonlinear equations // COAP. 2004. Vol. 28, No. 1. P. 31–50.
- [20] Gill P., Murray W., Wright M. Practical optimization. London: Academic Press, 1981.
- [21] Horst R., Tuy H. Global Optimization: Deterministic Approaches. Berlin: Springer-Verlag, 1996.
- [22] Strekalovsky A.S. On local search in d.c. optimization problems // Applied Mathematics and Computation. 2015. Vol. 255. P. 73–83.
- [23] Rockafellar R.T. Convex Analysis. Princeton, NJ: Princeton University Press, 1970.
- [24] IBM. IBM ILOG CPLEX manual. <http://www-03.ibm.com/software/products/ru/ibmilogcpleoptistud>. 2017. Accessed: 31.05.2017.
- [25] Gurobi Optimization, Inc. Gurobi optimizer reference manual. <http://www.gurobi.com>. 2016. Accessed: 31.05.2017.
- [26] Roose A., Kulla V., Lomp M., Meressoov T. Test examples of systems of non-linear equations Tallin: Estonian Software and Computer Service Company, 1990.

*Максим Викторович Янулевич — программист Института
динамики систем и теории управления имени В.М. Матросова СО РАН;
e-mail: max@irk.ru;*

*Александр Сергеевич Стрекаловский — д.ф.-м.н., зав. лабораторией Института
динамики систем и теории управления имени В.М. Матросова СО РАН;
e-mail: strekal@icc.ru.*

Дата поступления — 31 мая 2017 г.