

ЭФФЕКТИВНАЯ РЕАЛИЗАЦИЯ УСКОРЕННОГО МЕТОДА РЕШЕНИЯ ВАРИАЦИОННЫХ НЕРАВЕНСТВ

В.М. АЛЕКСАНДРОВА, Л.А. СОБОЛЕНКО

Построен нелокально сходящийся алгоритм решения вариационных неравенств с сильно монотонным оператором и выпуклыми ограничениями-неравенствами, обладающий высокой скоростью сходимости. Метод основан на совмещении глобального алгоритма первого порядка, использующего итерационную последовательность в пространстве прямых переменных, с методом Ньютона решения системы Куна-Таккера вариационных неравенств в окрестности решения. Выполнена эффективная реализация предложенного алгоритма. Рассмотрены вычислительные аспекты, связанные с двумя трудоемкими подзадачами сформулированного алгоритма — задачей квадратичного программирования и решением системы нелинейных равенств. Реализация метода опробована на решении вариационных неравенств с непотенциальным оператором. Проведен сравнительный анализ работы ускоренного алгоритма и алгоритма первого порядка. Высокая скорость сходимости предложенного алгоритма подтверждена результатами вычислительного эксперимента.

ВВЕДЕНИЕ

К вариационным неравенствам (в.н.) конечной размерности сводятся многие задачи математического программирования — задачи оптимизации, линейной и нелинейной дополнительной, отыскания седловых точек, которые возникают в экономических, транспортных, физических, технических и социальных исследованиях.

Наиболее полно численное решение в.н. конечной размерности представлено в [1–4]. Многие алгоритмы для решения в.н. основаны на решении последовательности линеаризованных в.н. на исходном нелинейном множестве. Они включают как методы первого порядка — проекционные методы, так и методы второго порядка — ньютоновского и квазиньютоновского типов. Вспомогательная задача этих алгоритмов состоит в минимизации линейного или квадратичного функционала на исходном множестве, поэтому практическое применение этих алгоритмов возможно в случае, когда множество имеет достаточно простую структуру, например, является многогранником. Кроме этого, недостатком этих методов является то, что все они, за исключением проекционных, сходятся к решению лишь с хорошего начального приближения.

Современным подходом к решению в.н. является построение эквивалентной оптимизационной задачи [5–7], для решения которой можно применить хорошо развитый аппарат математического программирования. Здесь основная трудность состоит в том, что эквивалентные оптимизационные задачи, как правило, сложны с вычислительной точки зрения. Они либо требуют перехода к целевым функциям большей размерности [5], либо к недифференцируемым функциям [1, 2], либо функциям, требующим вычисления проекции некоторой точки на исходное множество Ω [6].

В [7] построен глобальный алгоритм решения в.н. с непотенциальным оператором на основе эквивалентной оптимизационной задачи в исходном пространстве [8], решение которой, в силу ее специфики, можно заменить последовательным решением квадратичных подзадач в исходном пространстве в сочетании с одномерной минимизацией некоторой штрафной функции. Недостатком этого метода является то, что он обладает лишь линейной скоростью сходимости в окрестности решения.

Цель статьи — эффективное построение нелокально сходящегося алгоритма решения в.н., обладающего квадратичной скоростью сходимости в окрестности решения и использующего итерационный процесс по переменной в исходном пространстве.

ПОСТАНОВКА ЗАДАЧИ И ПРЕДВАРИТЕЛЬНЫЕ СВЕДЕНИЯ

Решение в. н. состоит в нахождении такой точки $x_* \in \Omega \subset \mathbf{R}^n$, что

$$\langle F(x_*), x - x_* \rangle \geq 0 \quad \forall x \in \Omega, \quad (1)$$

где $F(x) \in \mathbf{R}^n$, $x \in \mathbf{R}^n$, а $\langle \cdot, \cdot \rangle$ — обозначение скалярного произведения векторов \mathbf{R}^n .

Относительно оператора F и множества Ω будем предполагать, что выполняются следующие условия:

- оператор $F(x): \mathbf{R}^n \rightarrow \mathbf{R}^n$ непрерывно дифференцируем и сильно монотонен:

$$m \|p\|^2 \leq \langle F'(x)p, p \rangle \leq M \|p\|^2, \quad M \geq m > 0, \quad \forall x, p \in \mathbf{R}^n, \quad (2)$$

где матрица Якоби $F'(x)$ удовлетворяет условию Липшица, $\|\cdot\|$ — евклидова норма вектора в \mathbf{R}^n ;

- множество $\Omega \subset \mathbf{R}^n$ ограничено и задано в виде:

$$\Omega = \{x \in \mathbf{R}^n \mid h_i(x) \leq 0, i = 1, \dots, l\}, \quad (3)$$

где $h_i(x)$, $i = 1, \dots, l$ — выпуклые дважды непрерывно дифференцируемые функции такие, что матрицы вторых производных $h_i''(x)$, $i = 1, \dots, l$ удовлетворяют условию Липшица;

- градиенты $h_i'(x_*)$, $i \in I(x_*)$ линейно-независимы в решении задачи (1), где $I(x_*) = \{i \in 1, \dots, l \mid h_i(x_*) = 0\}$ — множество активных в точке x_* ограничений, а соответствующие ограничениям множители Лагранжа положительны: $\lambda_*^i > 0$, $i \in I(x_*)$ (условия регулярности ограничений).

Обозначим: $H(x)$ — $n \times l$ -матрица со столбцами $h_i'(x)$, $i = 1, \dots, l$, $\lambda \in \mathbf{R}^l$ — вектор множителей Лагранжа, $h(x) \in \mathbf{R}^l$ — вектор с компонентами $h_i(x)$, $i = 1, \dots, l$.

Задача (1) при условиях (2), (3) имеет единственное решение x_* , удовлетворяющее системе равенств:

$$\begin{aligned} F(x_*) + H(x_*)\lambda_* &= 0, \\ \lambda_*^i h_i(x_*) &= 0, \quad i = 1, \dots, l, \end{aligned} \quad (4)$$

а для активных в точке x_* ограничений следующей системе:

$$\begin{aligned} F(x_*) + \hat{H}(x_*)\hat{\lambda}_* &= 0, \\ \hat{h}(x_*) &= 0, \end{aligned} \quad (5)$$

где $\hat{H}(x_*)$ — $n \times |I(x_*)|$ -матрица; $\hat{\lambda}_*, \hat{h}(x_*) \in \mathbf{R}^{|I(x_*)|}$; $|I(x)|$ — количество элементов множества $I(x)$.

Поэтому решить исходную задачу (1)–(3) можно путем совмещения глобально сходящегося алгоритма для определения окрестности, в которой набор активных ограничений не меняется, с быстроходящимся алгоритмом решения системы равенств (5) в этой окрестности, например, методом Ньютона.

Итерация метода Ньютона решения системы (5) имеет вид:

$$\begin{aligned} &\begin{bmatrix} F'(x_k) + \sum_{i \in I(x_k)} \lambda_{k-1}^i h_i''(x_k) & \hat{H}(x_k) \\ \hat{H}^T(x_k) & 0 \end{bmatrix} \begin{bmatrix} x_{k+1} - x_k \\ \hat{\lambda}_{k+1} - \hat{\lambda}_k \end{bmatrix} = \\ &= - \begin{bmatrix} F(x_k) + \hat{H}(x_k)\hat{\lambda}_k \\ \hat{h}(x_k) \end{bmatrix}. \end{aligned} \quad (6)$$

В качестве глобально сходящегося метода решения (1)–(3) возьмем метод градиентного типа, сформулированный в [7]. Метод основан на переходе от в.н. (1) к эквивалентной оптимизационной задаче вида:

$$\min_{x \in \Omega} \Theta(x), \quad (7)$$

$$\Theta(x) = \min_{\lambda \in \Lambda^+} \left\{ \frac{1}{2} \|F(x) + H(x)\lambda\|^2 - \sum_{i=1}^l \lambda^i h_i(x) \right\}. \quad (8)$$

В силу специфики задачи (7), (8), ее решение сводится к последовательной минимизации функции $\varphi(x, \lambda) = \frac{1}{2} \|F(x) + H(x)\lambda\|^2 - \sum_{i=1}^l \lambda^i h_i(x)$ по переменным λ , а затем по переменным x . Минимизация по λ функции $\varphi(x, \lambda)$ является задачей квадратичного программирования, и ее решение $\lambda_k = \arg \min \varphi(x_k, \lambda)$, $\lambda \in \Lambda^+$ находится стандартными методами при фиксированной точке x_k . В силу двойственности переменных x и λ направление убывания $\varphi(x, \lambda)$ в пространстве переменных x определяется как $p_k = - \left(F(x_k) + \sum_{i=1}^l \lambda^i h_i'(x_k) \right)$ или из решения прямой задачи квадратичного программирования:

$$\min_{p \in \mathbf{R}^n} \left\{ \langle F(x_k), p \rangle + \frac{1}{2} \|p\|^2 |h_i(x_k) + \langle h'_i(x_k), p \rangle \leq 0, i = 1, \dots, l \right\}. \quad (9)$$

Вектор p_k , в свою очередь, является направлением убывания в точке x_k штрафной недифференцируемой функции $\Phi_N(x, \lambda_k) = \varphi(x, \lambda_k) + Nh^+(x)$, где $h^+(x) \equiv \max \{0, h_1(x), \dots, h_l(x)\}$, при условии, что $N \geq \sum_{i=1}^l \lambda_k^i$.

Для ускорения этого метода объединим его с методом Ньютона решения системы (5). Аналогичная проблема ускорения метода линеаризации в окрестности решения была решена для задач нелинейной условной оптимизации в [9, 10], а схема алгоритма для в.н. предложена в [11].

АЛГОРИТМ РЕШЕНИЯ

Сформулируем ускоренный алгоритм решения вариационных неравенств (1)–(3). Выберем произвольными начальное приближение $x_0 \in \mathbf{R}^n$, константу $\gamma \in (0; 1)$, точность решения $\varepsilon > 0$ — маленькое число; $\delta_0 > 0$, $C_0 > 0$, $N_0 > 0$ — достаточно большие числа. Обозначим множества $I_\delta(x) = \{i \in 1, \dots, l \mid h_i(x) \geq h^+(x) - \delta\}$, $I_\delta^0(x) = \{i \in I_\delta(x) \mid h_i(x) + \langle h'_i(x), p \rangle = 0\}$, $\delta > 0$.

Пусть x_k, C_k, δ_k, N_k для $k = 0, 1, \dots$ уже построены. Найдем x_{k+1} .

Шаг 1 (определение направления в прямом пространстве и двойственных переменных по градиентному методу). Решаем прямую (9) и двойственную (8) задачи квадратичного программирования в точке x_k для ограничений из множества $I_{\delta_k}(x_k)$. Пусть задачи (9), (8) имеют решения соответственно $p_k = p(x_k), \lambda_k^i = \lambda^i(x_k)$.

Если $\|p_k\| \leq \varepsilon$, то x_k — решение задачи (1). Конец алгоритма.

Если $\|p_k\| > C_k$, то полагаем $C_{k+1} = C_k$ и переходим к шагу 4.

Если $\|p_k\| \leq C_k$, то переходим к шагу 2.

Если задача (9) не имеет решения, то уменьшаем δ_k и повторяем шаг 1. За несколько дроблений δ_k либо находим решение задачи (9), либо убеждаемся, что ограничения вспомогательной задачи несовместны, а задача (1) не имеет решения. Конец алгоритма.

Шаг 2 (определение направления в прямом пространстве по методу Ньютона). Вычисляем d_k , решая систему уравнений:

$$\begin{bmatrix} F'(x_k) + \sum_{i \in I_{\delta_k}^0(x_k)} \lambda_{k-1}^i h_k''(x_k) & \hat{H}(x_k) \\ \hat{H}^T(x_k) & 0 \end{bmatrix} \begin{bmatrix} d \\ r \end{bmatrix} = - \begin{bmatrix} F(x_k) + \hat{H}(x_k) \hat{\lambda}_k \\ \hat{h}(x_k) \end{bmatrix}, \quad (10)$$

где $\hat{h}(x_k) \in \mathbf{R}^{|\mathcal{I}_{\delta_k^0}(x_k)|}$; $\hat{H}(x_k) — n \times |\mathcal{I}_{\delta_k^0}(x_k)|$ -матрица со столбцами $h'_i(x_k)$,
 $i \in \mathcal{I}_{\delta_k^0}(x_k)$, $d \in \mathbf{R}^n$, $r \in \mathbf{R}^{|\mathcal{I}_{\delta_k^0}(x_k)|}$.

Если система (10) не имеет решения, то полагаем $C_{k+1} = \gamma \|p_k\|$ и переходим к шагу 4. Если решение системы найдено, то переходим к шагу 3.

Шаг 3 (анализ приближения, полученного по методу Ньютона). Полагаем

$$\bar{x} = x_k + d_k. \quad (11)$$

В точке \bar{x} для $\bar{\delta} = \delta_k$ решаем задачу (9).

Если ограничения задачи (9) несовместны или в процессе ее решения потребовалось изменить $\bar{\delta}$, то полагаем $C_{k+1} = \gamma \|p_k\|$, $\delta_{k+1} = \delta_k$ и переходим к шагу 4.

Если задача (9) имеет решение $\bar{p} = p(\bar{x})$, вычисляем $\|\bar{p}\|$.

Если выполняется неравенство

$$\|\bar{p}\| \leq \gamma \|p_k\|, \quad (12)$$

то в качестве следующего приближения выбираем точку, полученную по методу Ньютона: полагаем $x_{k+1} = \bar{x}$, $C_{k+1} = \gamma \|\bar{p}\|$, $N_k = N_{k-1}$, $\delta_{k+1} = \delta_k$.
 Конец итерации.

Если неравенство (12) не выполняется, то полагаем $C_{k+1} = \gamma \|p_k\|$, $\delta_{k+1} = \delta_k$ и переходим к следующему шагу.

Шаг 4 (вычисление нового приближения по градиентному методу).

Определяем коэффициент штрафа. Пусть $Y(x_k) = \sum_{i \in \mathcal{I}_{\delta}(x_k)} |\lambda_k^i|$. Если $N_{k-1} < Y(x_k)$, то полагаем $N_k = 2Y(x_k)$, если $N_{k-1} \geq Y(x_k)$, то полагаем $N_k = 2^{-\nu} N_{k-1}$, $\nu = 1, \dots$ до тех пор, пока не выполнится условие $N_{k-1} < Y(x_k)$, тогда $N_k = 2N_{k-1}$. Коэффициент штрафа стабилизируется в пределах $Y(x_k) \leq N_k \leq 2Y(x_k)$.

Выбираем шаговый множитель вдоль направления p_k из релаксации штрафной функции $\Phi_{N_k}(x, \lambda_k) = \varphi(x, \lambda_k) + N_k h^+(x)$, где $h^+(x) \equiv \max\{0, h_1(x), \dots, h_l(x)\}$. Начиная с $\alpha = 1$, дробим его путем деления пополам до выполнения условия:

$$\Phi_{N_k}(x_k + \alpha p_k, \lambda_k) \leq \Phi_{N_k}(x_k, \lambda_k) - \varepsilon \alpha^2 \|p_k\|^2,$$

находим α_k . Полагаем $x_{k+1} = x_k + \alpha_k p_k$. Конец итерации.

При сделанных предположениях относительно оператора $F(x)$, функций ограничений $h_i(x)$, $i = 1, \dots, l$, матриц $F'(x)$ и $h'_i(x)$, $i = 1, \dots, l$ в окрест-

ности решения x_* матрица системы (10) не вырождена, а переход от x_k к x_{k+1} осуществляется по формуле (11), что гарантирует квадратичную скорость сходимости итерационной последовательности [9, 10].

ВЫЧИСЛИТЕЛЬНЫЕ АСПЕКТЫ

Двумя трудоемкими подзадачами сформулированного алгоритма являются задача квадратичного программирования (9) и решение системы (10). Очевидно, что от эффективности их решения во многом зависит эффективность решения исходной задачи. Поэтому рассмотрим способы решения и вычислительные аспекты, возникающие в процессе их решения, более подробно.

Метод, которым будем решать задачу (9), относится к семейству методов активного набора. Основная идея методов этого типа состоит в следующем. Задача квадратичного программирования с ограничениями-неравенствами заменяется на последовательное решение подзадач с ограничениями-равенствами. Для этого вводится рабочий список — список ограничений задачи (9), интерпретируемых в процессе поиска решения как равенства. В него будет входить только часть ограничений. Эти ограничения определяют некоторую грань многогранного множества, описываемого линейными неравенствами задачи (9). Находим минимум на этой грани. Полученная точка либо является решением задачи (9), если множество ограничений из рабочего списка совпадет с множеством активных ограничений в решении, либо возможен переход на некоторую новую грань, после чего процедура поиска повторяется.

Среди методов активного набора выделяют два направления — методы прямого и двойственного активного набора [12]. В методах первого типа итерационная последовательность начинается и все время удерживается в допустимой области, при этом от итерации к итерации значение целевой функции уменьшается, пока не достигнет минимума.

Итерационная последовательность методов двойственного активного набора начинается с недопустимой точки безусловного минимума квадратичной функции и, увеличивая значение целевой функции, последовательно удовлетворяет всем ограничениям задачи. Второй способ предпочтительней, поскольку нахождение безусловного минимума в случае выпуклой квадратичной функции более просто, чем нахождение какой-нибудь допустимой точки множества.

В численных экспериментах, которые проводились авторами, вспомогательная задача (9) решалась двойственным методом активного набора [13]. В этом алгоритме предусмотрено одновременное вычисление решения по прямым и двойственным переменным, что требует решения систем линейных уравнений. Для решения систем уравнений во вспомогательных задачах использовались матричные преобразования — разложение Холецкого и QR -факторизация, с помощью которых матрицы приводились к треугольному виду. Решение систем уравнений таким способом, в отличие от процедур исключения Гаусса, является численно устойчивым [12].

Напомним, что разложение Холецкого позволяет симметричную положительно определенную $n \times n$ -матрицу G представить в виде: $G = LL^T$, где L — нижняя треугольная матрица размерности $n \times n$. Кроме того, если диагональные элементы L выбираются положительными, то разложение единственно. Элементы L можно определять по строкам и по столбцам. Если матрицу L определять по строкам, то элементы i -й строки вычисляются из следующих соотношений:

$$l_{ij} = \frac{g_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}}{l_{jj}}, \quad j=1, \dots, i-1, \quad l_{ii} = \left(g_{ii} - \sum_{k=1}^{i-1} l_{ik}^2 \right)^{1/2}, \quad i=1, \dots, n.$$

Ортогонализация (QR -факторизация) — это преобразование произвольной $n \times q$ -матрицы B ранга q к верхнетреугольному виду, т.е. $QB = \begin{bmatrix} R_q \\ 0 \end{bmatrix}$, где R_q — верхняя треугольная матрица размерности $q \times q$, $Q = P_q \cdot P_{q-1} \cdot \dots \cdot P_1$ — ортонормальная $n \times n$ -матрица, определяемая элементарными матрицами Хаусхолдера P_1, \dots, P_q .

Матрица Хаусхолдера — матрица вида: $P = E_n - \frac{2vv^T}{\|v\|^2}$, где $v \in \mathbf{R}^n$,

E_n — единичная $n \times n$ -матрица. Из определения матрицы P следует, что она воздействует на векторы с сохранением евклидовой нормы, т.е. $\|Pb\| = \|b\|$. Для двух не совпадающих векторов $b, \bar{b} \in \mathbf{R}^n$ таких, что $\|b\| = \|\bar{b}\|$, всегда можно подобрать преобразование Хаусхолдера P , которое преобразует один вектор в другой, т.е. $Pb = \left(E_n - \frac{2vv^T}{\|v\|^2} \right) b = b - \frac{2\langle v, b \rangle}{\|v\|^2} v = \bar{b}$. Из последнего соотношения следует, что, если $\langle v, b \rangle = 0$, то $Pb = b$, кроме того, если для некоторых i таких, что $v_i = 0$, то $b_i = \bar{b}_i$.

Для преобразования матрицы B к верхней треугольной матрица P_1 подбирается так, чтобы 1-й столбец матрицы B преобразовать в вектор $\left(\sqrt{\sum_{i=1}^n b_{i1}^2}, 0, \dots, 0 \right)^T$, т.е. обнулить все компоненты, кроме первой. Соответственно, P_i подбирается так, чтобы в i -м столбце преобразованной матрицы B обнулить компоненты с индексами $i+1, \dots, n$ не меняя компоненты с индексами $1, \dots, i-1$.

Среди ортогональных преобразований выделяют плоские повороты Гивенса. Поворот Гивенса используется для обнуления какой-нибудь одной компоненты вектора b и задается матрицей Хаусхолдера с вектором v , у которого только две ненулевые компоненты $(0 \dots v_i \dots v_j \dots 0)^T$. Матрица Гивенса, соответствующая плоскому повороту, имеет вид:

$$Q_{(i,j)} = \begin{matrix} & i & & j & \\ & & & & \\ & & & & \\ i & & & & \\ & & & & \\ & & & & \\ j & & & & \\ & & & & \\ & & & & \end{matrix} \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & c & 0 & s & 0 \\ \vdots & 0 & 1 & 0 & \vdots \\ 0 & -s & 0 & c & 0 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix},$$

где $c^2 + s^2 = 1$, $c = \cos \alpha$, $s = \sin \alpha$, для некоторого угла α . Умножение вектора b слева на рассматриваемую матрицу $Q_{(i,j)}$ изменит только i -ю и j -ю компоненты вектора. Угол α подбирают так, чтобы обнулить j -ю компоненту вектора b , например, чтобы преобразовать вектор $b = (0, \dots, b_i, \dots, \dots, b_j, \dots, 0)^T$ в вектор $Q_{(i,j)}b = (0, \dots, r_i, \dots, 0, \dots, 0)^T$, где $r_i = \pm(b_i^2 + b_j^2)^{1/2}$.

Суть процедуры рассмотрим на примере матрицы Гивенса, размерности 2×2 . Чтобы определить параметры ортогонального преобразования

$$q = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}, \quad q \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} r_1 \\ 0 \end{bmatrix}, \quad r_1 = \pm(b_1^2 + b_2^2)^{1/2},$$

необходимо вычислить

$$\mu = \max\{|b_1|, |b_2|\}, \quad r_1 = \text{sign}(b_1)\mu \left(\left(\frac{b_1}{\mu} \right)^2 + \left(\frac{b_2}{\mu} \right)^2 \right)^{1/2}, \quad c = \frac{b_1}{r_1}, \quad s = \frac{b_2}{r_1}.$$

Плоские повороты Гивенса в алгоритме решения вспомогательной задачи квадратичного программирования (9) используются для пересчета QR -факторов матриц систем уравнений при добавлении ограничения в рабочий список или при удалении ограничения из него.

Рассмотрим пересчет QR -факторов на примере матрицы B . Рассмотрим добавление $q+1$ -го столбца в факторизованную матрицу B , т.е. пусть

$$QB = \begin{bmatrix} R_q & b_1 \\ 0 & b_2 \end{bmatrix}. \quad \text{Чтобы получить треугольную матрицу } R_{q+1}, \text{ необходимо}$$

преобразовать вектор $b_2 \in R^{n-q}$ в вектор $\begin{bmatrix} \delta \\ 0 \end{bmatrix}$, где $\delta = \left(\sum_{i=1}^{n-q} b_{2_i}^2 \right)^{1/2}$, т.е. необ-

ходимо обнулить в векторе $b_2 \in R^{n-q}$ все компоненты, кроме первой. Это можно сделать либо одним ортонормальным преобразованием Хаусхолдера, либо последовательным применением матриц плоского поворота Гивенса, обнуляя компоненты вектора $b \in R^n$, начиная с n -й компоненты. Обозначим \tilde{Q} — произведение матриц Гивенса, соответствующее этим преобразованиям. В результате имеем:

$$\tilde{Q} = \begin{bmatrix} E_q & 0 \\ 0 & \tilde{Q}_2 \end{bmatrix}; \quad \tilde{Q}QB = \begin{bmatrix} E_q & 0 \\ 0 & \tilde{Q}_2 \end{bmatrix} \begin{bmatrix} R_q & d_1 \\ 0 & d_2 \end{bmatrix} = \begin{bmatrix} R_q & d_1 \\ 0 & \delta \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R_{q+1} \\ 0 \end{bmatrix},$$

где $\tilde{Q}_2 = Q_{((q+2),(q+1))} \cdots Q_{((n-1),(n-2))} Q_{(n,(n-1))}$, R_{q+1} — треугольная $(q+1) \times (q+1)$ -матрица, E_q — единичная $q \times q$ -матрица.

Если l -й столбец вычеркивается из преобразованной матрицы B , то $QB = \begin{bmatrix} K \\ 0 \end{bmatrix}$, где K — матрица размерности $q \times (q-1)$ вида $K = \begin{bmatrix} R_{l-1} & S \\ 0 & T \end{bmatrix}$, R_{l-1} — треугольная матрица размерности $(l-1) \times (l-1)$, T — верхний Хессенберг размерности $(q-l+1) \times (q-l)$. Поддиагональные элементы матрицы T можно обнулить последовательностью $(q-l)$ матриц плоского поворота. Обозначим \bar{Q} — произведение матриц Гивенса, соответствующее этим преобразованиям. Тогда

$$\bar{Q} = \begin{bmatrix} E_{l-1} & 0 & 0 \\ 0 & \bar{Q}_3 & 0 \\ 0 & 0 & E_{n-q} \end{bmatrix}, \quad \bar{Q}QB = \bar{Q} \begin{bmatrix} R_{l-1} & S \\ 0 & T \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R_{q-1} \\ 0 \end{bmatrix},$$

где $\bar{Q}_3 = Q_{((n-q-l),(n-q-l+1))} \cdots Q_{((n-q-1),(n-q-2))} Q_{((n-q),(n-q-1))}$, R_{q-1} — треугольная, E_{l-1} , E_{n-q} — единичные матрицы соответствующих размерностей.

ЧИСЛЕННЫЙ ЭКСПЕРИМЕНТ

Прикладные модули предложенного в данной статье ускоренного алгоритма, реализованы в виде классов на языке C# для платформы NET 3.5, которые включают: класс для решения вспомогательной квадратичной задачи (9), класс для решения системы (10) методом Ньютона, класс для решения в.н. методом [7], вспомогательные классы для задания входных параметров и функций задачи, класс для вывода результатов решения, класс для графического представления данных в процессе решения и, собственно, класс для ускоренного алгоритма. Как и ожидалось, предложенный алгоритм дает значительное ускорение, по сравнению с алгоритмом первого порядка [7]. Работа алгоритма тестировалась на примерах из работ [14, 15].

На рисунке представлены графики зависимости величины $\|p\|$ от номера итерации для метода [7] и предложенного ускоренного алгоритма на примере решения задачи из [15]:

$$F(x) = \begin{bmatrix} 3x_1^2 + 2x_1x_2 + 2x_2^2 + x_3 + 3x_4 - 6 \\ 2x_1^2 + x_1 + x_2^2 + 3x_3 + 2x_4 - 2 \\ 3x_1^2 + x_1x_2 + 2x_2^2 + 2x_3 + 3x_4 - 1 \\ x_1^2 + 3x_2^2 + 2x_3 + 3x_4 - 3 \end{bmatrix},$$

$\Omega = \{x_i \geq 0, i = 1, \dots, 4\}$, начальное приближение — точка $x_0 = (1, 1, 1, 1)^T$, решение — точка $x_* = (0,5\sqrt{6}; 0; 0; 0,5)^T$.

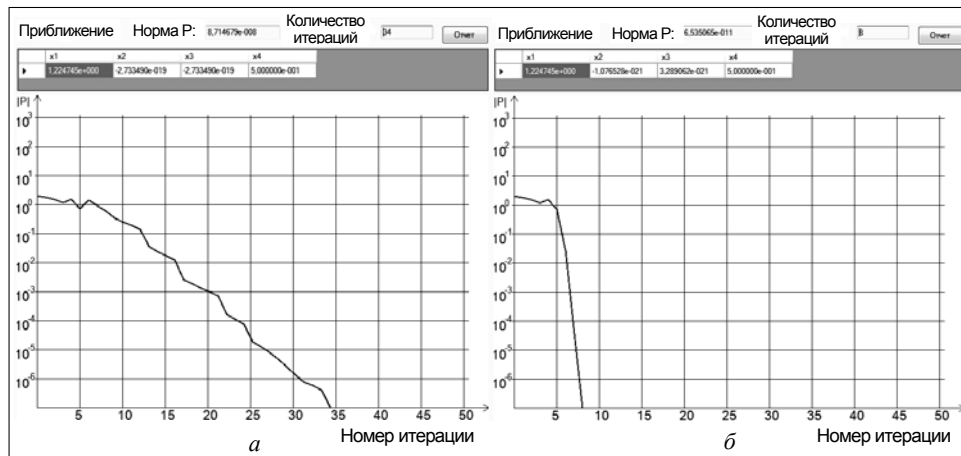


Рисунок. Сравнение сходимостей рассматриваемых методов: a — сходимость метода первого порядка, b — сходимость ускоренного метода

Стремление к нулю нормы вектора p является признаком сходимости итерационной последовательности алгоритма к решению задачи (1) по переменной x . Как показал анализ работы алгоритмов, ускоренный алгоритм получает решение задачи за 8 итераций (7 итераций по методу первого порядка [7] и одна итерация по методу Ньютона), в то время, как методу первого порядка для достижения решения с заданной точностью понадобилось 34 итерации.

В представленной алгоритмической схеме решение двух вспомогательных задач предусмотрено с 1-й итерации, но решение системы равенств на шаге 2 и последующую проверку полученного решения на шаге 3 можно исключить на начальных итерациях путем введения дополнительного управляющего параметра [10].

ВЫВОДЫ

Рассмотрены конечномерные выпуклые в.н. с непотенциальным оператором. Предложена алгоритмическая схема построения последовательности приближений x_k , которая сходится к решению в.н. с любого начального приближения с квадратичной скоростью. Результирующая итерационная последовательность строится на основании последовательностей точек, получаемых двумя методами — глобальным методом первого порядка, предложенного в [7], и локальным методом решения системы необходимых условий в.н. в окрестности решения. Выбор очередного приближения контролируется применением управляющей последовательности C_k , которая не позволяет выбирать приближение, полученное по методу Ньютона, если это приводит к увеличению $\|p_k\|$.

Все рассматриваемые в работе алгоритмы реализованы в виде классов C# для платформы NET 3.5. Реализации вспомогательных задач — задачи квадратичного программирования и решения системы уравнений методом Ньютона, включают численно устойчивые процедуры матричных разложе-

ний. В работе кратко изложена суть используемых матричных разложений — представление симметричной положительно определенной матрицы в виде произведения двух треугольных матриц (разложение Холецкого), ортонормальное преобразование произвольной матрицы и приведение ее к верхнетреугольному виду (QR -факторизация), а также пересчет факторов разложений матриц с помощью плоских поворотов Гивенса.

Представлены результаты тестирования реализованных алгоритмов на решении в.н. с несимметричной матрицей Якоби. Численные эксперименты и сравнительный анализ алгоритмов, наглядно подтверждают высокую скорость сходимости предложенного алгоритма.

ЛИТЕРАТУРА

1. *Facchinei F., Pang J.-S.* Finite-Dimensional Variational Inequalities and Complementarity, Volume I. — Springer, 2003. — 728 p.
2. *Facchinei F., Pang J.-S.* Finite-Dimensional Variational Inequalities and Complementarity, Volume II. — Springer, 2003. — 702 p.
3. *Xiao B., Harker P.T.* A nonsmooth Newton method for variational inequalities, I: Theory // *Math. Progr.* — 1994. — **65**, № 2. — P. 151–194.
4. *Бакушинский Я. М., Поляк Б. Т.* О решении вариационных неравенств // Докл. АН СССР. — 1974. — **219**, № 5, — С. 1038–1041.
5. *Панин В.М., Александрова В.М.* Нелокальный квазиньютоновский метод решения выпуклых вариационных неравенств // Кибернетика и систем. анализ. — 1994. — № 6. — С. 78–91.
6. *Fukushima M.* Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems // *Math. Progr.* — 1992. — **53**, № 2. — P. 99–110.
7. *Александрова В.М., Шубенкова И.А.* Об усовершенствовании метода решения вариационных неравенств // Кибернетика и системный анализ. — 2013. — № 2. — С. 150–156.
8. *Данилин Ю.М., Шубенкова И.А.* Оптимизация и решение выпуклых вариационных неравенств // Системні дослідження та інформаційні технології. — 2003. — № 3. — С. 66–74.
9. *Пшеничный Б.Н.* Метод линеаризации. — М.: Наука, 1983. — 136 с.
10. *Соболенко Л.А.* Ускоренные модификации метода линеаризации. — Деп. ВИНТИ 7525. — В.89. — 1989. — 35 с.
11. *Александрова В.М.* Ускоренный метод решения вариационных неравенств // Теория оптимальных решений. — Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 1995. — С. 8–13.
12. *Гилл Ф., Мюррей К., Райт М.* Практическая оптимизация. — М.: Мир, 1985. — 509 с.
13. *Goldfarb D, Idnani A.* A numerically stable dual method for solving strictly convex quadratic problem // *Math. Progr.* — 1983. — **27**, № 1. — P. 1–33.
14. *Fukushima M.* A relaxed projection method for variational inequalities // *Math. Progr.* — 1986. — **35**, № 1. — P. 58–70.
15. *Taji K., Fukushima M., Ibaraki T.* A globally convergent Newton method for solving strongly monotone variational inequalities // *Math. Progr.* — 1993. — **58**, № 3. — P. 369–383.

Поступила 21.03.2013