

Math-Net.Ru

Общероссийский математический портал

И. Г. Черноруцкий, Градиентные методы оптимизации больших систем,
*Научно-технические ведомости СПбГПУ. Информатика. Телекоммуника-
ции. Управление*, 2014, выпуск 4, 47–56

<https://www.mathnet.ru/ntitu48>

Использование Общероссийского математического портала Math-Net.Ru подразумевает, что вы прочита-
ли и согласны с пользовательским соглашением

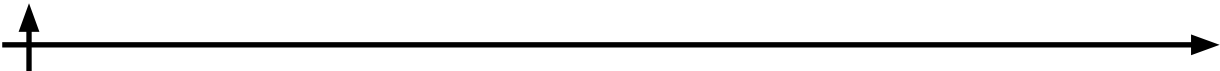
<https://www.mathnet.ru/rus/agreement>

Параметры загрузки:

IP: 18.97.9.174

25 апреля 2025 г., 00:05:18





Моделирование вычислительных, телекоммуникационных, управляющих и социально-экономических систем

УДК 681.3.06

И.Г. Черноруцкий

ГРАДИЕНТНЫЕ МЕТОДЫ ОПТИМИЗАЦИИ БОЛЬШИХ СИСТЕМ

I.G. Chernorutskiy

GRADIENT METHODS FOR LARGE-SCALE MINIMIZATION PROBLEMS

Разработаны градиентные методы с чебышевскими функциями релаксации. В отличие от классических градиентных процедур, построенные методы сохраняют сходимость и эффективность для невыпуклых задач нелинейного программирования в условиях высокой степени жесткости целевых функционалов и высокой размерности вектора оптимизируемых параметров.

ГРАДИЕНТНЫЕ МЕТОДЫ; ФУНКЦИИ РЕЛАКСАЦИИ; НЕВЫПУКЛЫЕ ЗАДАЧИ; ЖЕСТКИЕ ФУНКЦИОНАЛЫ.

Gradient methods with Chebyshev relaxation functions are developed. In contrast to the classical gradient procedures, the methods retain the convergence and efficiency for non-convex nonlinear programming problems under the conditions of high stiffness of target functionals and high dimension of the optimized parameters vector.

GRADIENT METHODS; RELAXATION FUNCTIONS; NON-CONVEX PROBLEMS; STIFF FUNCTIONALS.

Для решения задачи безусловной минимизации

$$J(x) \rightarrow \min_x, x \in R^n, J \in C^2(R^n)$$

рассматривается класс матричных градиентных методов вида

$$x^{k+1} = x^k - H_k(G_k, h_k)J'(x^k), \quad h_k \in R^1, \quad (1)$$

где $G_k = J''(x^k)$, H_k – матричная функция G_k .

Данный класс методов, как показано в работах [1–8], включает в себя как частные случаи такие классические процедуры, как градиентные методы наискорейшего спуска, методы Левенберга-Маркуардта, ньютоновские методы, методы с экспоненциальной релаксацией (ЭР-методы).

Далее представлен основанный на понятии функции релаксации [1, 6] подход к построению и анализу нетрадиционных градиентных методов с чебышевской функ-

цией релаксации. Основные предположения, используемые при построении данного класса методов:

- высокая размерность вектора аргумента x ($n \gg 1$) и нежелательность хранения полноразмерной матрицы ($n \times n$) в памяти компьютера;

- высокая степень жесткости [1, 4, 5] функционала $J(x)$ в широкой области изменения аргумента;

- выпуклость минимизируемого функционала $J(x)$ гарантируется только в окрестности точки минимума.

Задачи многопараметрической оптимизации

Под большими системами будем понимать системы, описываемые моделями с большим числом управляемых параметров. Если степень жесткости соответствующих критериев оптимальности достаточно высока, то стандартные вычислительные

средства оказываются неэффективными в силу изложенных в [1, 4, 5] причин. Методы с экспоненциальной релаксацией [8] неприменимы, т. к. их вычислительные схемы содержат заполненные матрицы размерности $n \times n$, что при больших (порядка 1000) n определяет чрезмерные требования к объему необходимой памяти компьютера. Методы ньютоновского типа, как показано в [8], не предназначены для решения невыпуклых задач и, кроме того, теряют эффективность в условиях высокой степени жесткости.

Наиболее часто в указанной ситуации рекомендуется применять различные нематричные формы метода сопряженных градиентов (СГ). Однако далее будет показано, что в классе матричных градиентных схем (1) существуют более эффективные для рассматриваемых задач алгоритмы, чем методы СГ.

Пусть оптимизируемая система может быть представлена как совокупность взаимосвязанных подсистем меньшей размерности. Пусть также требования к выходным параметрам системы (спецификации) могут быть сформулированы в виде следующих неравенств:

$$y_j(x_j, x^q) \leq t_j, j \in [1: q - 1]; y_q(x^q) \leq t_q, \quad (2)$$

где x^j есть n_j -мерный частный вектор управляемых параметров; $x^q - n_q$ -мерный вектор управляемых параметров, влияющий на все q выходных параметров и осуществляющий связь отдельных подсистем оптимизируемой системы. Размерность полного вектора управляемых параметров $x = [x^1, x^2, \dots, x^q]$ равна

$$n = \sum_{i=1}^q n_i. \quad (3)$$

Используя технику оптимизации, представленную в [1], можно привести задачу решения системы неравенств (1) к виду

$$J(x) = \sum_{j=1}^q \Psi_j(x^j, x^q) \rightarrow \min, x \in R^n, \quad (4)$$

что является сглаженным вариантом критерия минимального запаса работоспособности.

Функционалы (4) возникают и при других постановках задач оптимального пара-

метрического синтеза, не основанных непосредственно на критериях минимального запаса работоспособности. Поэтому задача (4) имеет достаточно общий характер.

Далее будут рассмотрены методы решения задачи (4) при следующих дополнительных предположениях:

- решение задачи анализа оптимизируемой системы требует значительных вычислительных затрат, поэтому в процессе оптимизации требуется минимизировать количество обращений к вычислению значений $J(x)$;

- коэффициент заполнения γ матрицы $G(x) = J''(x)$ достаточно мал. Обычно можно полагать $\gamma \sim 1/q$.

Легко установить, что структура матрицы $G(x)$ не зависит от точки x :

$$G(x) = \begin{bmatrix} G_{11} & 0 & G_{1q} \\ & G_{22} & G_{2q} \\ 0 & & G_{33} & G_{3q} \\ & & & \ddots & \vdots \\ G_{q1} & G_{q2} & G_{q3} & \dots & G_{qq} \end{bmatrix}.$$

Подматрицы G_{ij} имеют размеры $n_i \times n_j$, а общее число ненулевых элементов равно

$$\sum_{i=1}^q n_i^2 + 2n_q \sum_{i=1}^{q-1} n_i.$$

Таким образом, учитывая симметричность матрицы $G(x)$, в памяти компьютера необходимо хранить

$$\sum_{i=1}^q (n_i^2 + n_i)/2 + n_q \sum_{i=1}^{q-1} n_i$$

ненулевых элементов. Необходимые сведения о схемах хранения разреженных матриц содержатся, например, в [9, 10].

Методы с чебышевскими функциями релаксации

Пусть $\lambda_i(G_k) \in [-m, M]$, $M \gg m > 0$. В силу приведенных выше предположений и сформулированных в [6] требований к функциям релаксации, наиболее рациональный метод должен иметь функцию релаксации, значения которой резко снижаются от $R = 1$ при $\lambda = 0$, оставаясь малыми во всем диапазоне $[0, M]$. И напротив, при $\lambda < 0$ функция $R(\lambda)$ должна

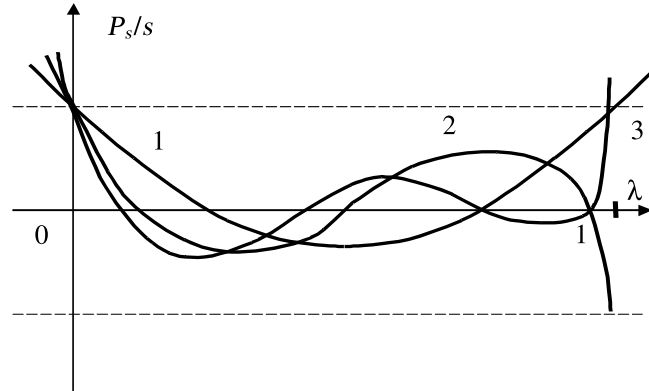


Рис. 1. Чебышевские функции релаксации

интенсивно возрастать. Кроме того, отвечающая $R(\lambda)$ матричная функция H должна строиться без матричных умножений для сохранения свойства разреженности матрицы $G_k = J''(x^k)$.

Покажем, что в качестве такой $R(\lambda)$ с точностью до множителя могут быть использованы смещенные полиномы Чебышева второго рода $P_s(\lambda)$, удовлетворяющие следующим соотношениям [11]:

$$\begin{aligned} P_1(\lambda) &= 1, P_2(\lambda) = 2(1 - 2\lambda); \\ P_{s+1}(\lambda) &= 2(1 - 2\lambda)P_s(\lambda) - P_{s-1}(\lambda). \end{aligned} \quad (5)$$

Графики зависимостей $P_s(\lambda)/s$ для $s = 3, 4, 5$ представлены на рис 1. Действительно, полагая $R(\lambda) = P_L(\lambda)/L$ при достаточно большом значении L получим сколько угодно быструю релаксацию любого слагаемого в представлении

$$f(x^{k+1}) = \frac{1}{2} \sum_{i=1}^n \xi_{i,k}^2 \lambda_i R^2(\lambda_i), \quad (6)$$

где

$$x^k = \sum_{i=1}^n \xi_{i,k} u^i.$$

Данное утверждение вытекает из известного факта равномерной сходимости последовательности $\{P_s(\lambda)/s\}$ к нулю при $s \rightarrow \infty$ на открытом промежутке $(0, 1)$. Далее будем предполагать, что собственные числа матрицы G_k нормированы к промежутку $(0, 1)$. Для этого достаточно вместо матрицы G рассматривать матрицу $G/\|G\|$, а вместо вектора градиента g — вектор $g/\|G\|$.

Отвечающая принятой $R(\lambda)$ зависимость $H(\lambda)$ имеет вид

$$H(\lambda) = [1 - R(\lambda)]/\lambda = [1 - P_L(\lambda)/L]/\lambda. \quad (7)$$

Построение методов (1) непосредственно с функцией (7) возможно, но приводит к необходимости решения на каждом шаге по k больших линейных систем уравнений с разреженной матрицей. Ниже показано, что существуют более эффективные приемы реализации.

Из (7) следует, что $H(\lambda)$ является полиномом степени $L - 2$, в то время как $R(\lambda)$ имеет степень $L - 1$. Поэтому для реализации матричного градиентного метода с указанной функцией $H(\lambda)$, вообще говоря, нет необходимости решать линейные системы. Метод будет выглядеть следующим образом:

$$\begin{aligned} x^{k+1} &= x^k - (\alpha_1 E + \alpha_2 G_k + \dots + \alpha_{L-1} G_k^{L-2}) g^k = \\ &= x^k - H(G_k) g^k. \end{aligned} \quad (8)$$

Реализация метода (8) может быть основана на методах вычисления коэффициентов α_i для различных степеней L . При этом число L должно выбираться из условия наиболее быстрого убывания $J(x)$. Далее обсуждается альтернативный подход, основанный на других соображениях.

Для функции

$$H_s(\lambda) \triangleq \alpha_1 + \alpha_2 \lambda + \dots + \alpha_{s-1} \lambda^{s-2}, \quad s = 2, 3, \dots$$

из (5) можно получить рекуррентное соотношение

$$\begin{aligned} (s+1)H_{s+1} &= 2s(1 - 2\lambda)H_s - (s-1) \times \\ &\times H_{s-1} + 4s; \end{aligned}$$

$$H_1 = 0, H_2 = 2, s \in [2: L - 1]. \quad (9)$$

Отсюда имеем

$$x^{k+1}[s+1] \triangleq x^k - H_{s+1}g^k =$$

$$= x^k - \frac{2s}{s+1}(E - 2G_k)H_s g^k + \frac{s-1}{s+1}H_{s-1}g^k -$$

$$-\frac{4s}{s+1}g^k, s \in [2 : L-1]$$

или

$$\vartheta_{s+1} \triangleq x^{k+1}[s+1] - x^k =$$

$$= \frac{2s}{s+1}(E - 2G_k)\vartheta_s - \frac{s-1}{s+1}\vartheta_{s-1} - \frac{4s}{s+1}g^k; (10)$$

$$\vartheta_1 = 0, \vartheta_2 = -2g^k, s \in [2 : L-1].$$

Здесь $x^{k+1}[s]$ есть s -е приближение к вектору $x^{k+1} = x^{k+1}[L]$.

Таким образом, при фиксированной квадратичной аппроксимации $f(x)$ функционала $J(x)$ в окрестности $x = x^k$, мы имеем

| | | | | | | | |
|------------|-----|-------|-------|-------|-------|-------|-------|
| s | ... | 3 | 4 | 5 | 6 | 7 | 8 |
| R_s | ... | 0,333 | 0,272 | 0,250 | 0,239 | 0,233 | 0,230 |
| α_s | ... | 0,147 | 0,092 | 0,061 | 0,044 | 0,033 | 0,025 |
| β_s | ... | 0,853 | 0,908 | 0,939 | 0,956 | 0,967 | 0,975 |
| $-R'_s(0)$ | ... | 5,30 | 10,0 | 16,0 | 23,3 | 32,0 | 42,0 |

В левой части спектра ($\lambda < 0$) имеем $R_s(\lambda) > 1 + R'_s(0)\lambda$, поэтому значения производных $R'_s(0)$ в последней строке таблицы характеризуют множители релаксации для отрицательных слагаемых в (6). Вычисление производных $R'_s(0)$ может быть выполнено, исходя из следующих рекуррентных соотношений:

$$P'_1 = 0, P'_2 = -4; P'_{s+1} =$$

$$= 2P'_s - 4s - P'_{s-1}; R'_L(0) = P'_L/L.$$

Значения α_s, β_s для $s > 8$ (при $\lambda > 0$) могут быть вычислены по асимптотической формуле:

$$\alpha_s = 1,63/s^2, \beta_s = 1 - \alpha_s; (11)$$

при этом $R_s < 0,22$.

Соотношение (11) получается из следующего представления полиномов Чебышева:

$$P_L(\lambda) = \frac{\sin L\zeta}{L \sin \zeta}, \lambda = \sin^2 \frac{\zeta}{2}, \lambda, \zeta \in [0, 1].$$

Действительно, при достаточно малых ζ

возможность переходить от P_s к P_{s+1} за счет одного умножения матрицы $E - 2G_k$ на вектор ϑ_s , в полной мере используя свойство разреженности матрицы G_k и не прибегая к дополнительным вычислениям градиента. Эффективность алгоритма (9) при больших значениях η определяется множителями релаксации для малых собственных значений матрицы G_k . Рассмотрим положительную часть спектра ($\lambda > 0$), что особенно важно в окрестности оптимума, где матрица $G(x)$ положительно определена. Основное достоинство метода с $R_s(\lambda) = P_s(\lambda)/s$ состоит в том, что уже при малых s происходит заметное подавление слагаемых из (5) в широком диапазоне значений λ . Ниже представлены значения R_s для внутреннего максимума $R_s(\lambda)$ и границы диапазонов $\alpha_s \leq \lambda \leq \beta_s$, где $|R'_s(\lambda)| \leq R_s$:

имеем [12]:

$$P_L(\lambda) \cong \Phi(\xi) = \frac{\sin \sqrt{\xi}}{\sqrt{\xi}}, \xi \triangleq 4L^2\lambda.$$

Полагая $x = \sqrt{\xi}$, получим $\Phi(\xi) = \varphi(x) = \sin x/x$. Имеем $\Phi(\xi) \leq \Phi(\xi_{кр})$ при $\xi \geq \xi_{кр}$, где $\xi_{кр} = x_{кр}^2 = 6,523$; $\Phi(\xi_{кр}) = \varphi(x_{кр}) \cong 0,22$.

Таким образом, полагая $\xi_{кр} = 4L^2\lambda_{кр}$, получим следующее утверждение: если для наименьшего (положительного) собственного числа m выполняется неравенство

$$\xi_{\min} = 4L^{2m} \geq \xi_{кр} = 6,523,$$

то есть если

$$m \geq 6,523/(4L^2) = 1,63/L^2, (12)$$

то для всех $\lambda > m$ будем иметь

$$|R_L(\lambda)| \leq 0,22.$$

Из (12) следует (11).

Узкая схема алгоритма, построенного на основе соотношения (10), может быть реализована с помощью следующей последовательности шагов. При

этом предполагается, что все переменные задачи надлежащим образом нормализованы. Предполагается также, что переменные пронумерованы некоторым оптимальным способом, гарантирующим эффективное хранение разреженной матрицы $(E - G_k)$ в памяти компьютера.

Алгоритм RELCH

Шаг 1. Задать начальную точку x ; вычислить $J := J(x)$; задать L , определяющее количество пересчетов по формуле (10) (об априорном выборе L см. ниже).

Шаг 2. Вычислить $g := J'(x)$, $G := J''(x)$; положить $g := g / \|G\|$, $G := G / \|G\|$; $\alpha := 1$.

Шаг 3. По формуле (10) построить ϑ_L ; положить $x^t := x + \vartheta_L$.

Шаг 4. Вычислить $J_t := J(x^t)$. Если $J_t > J$, перейти к шагу 5, иначе – к шагу 6.

Шаг 5. Положить $\alpha := \alpha/2$, $x^t := x + \alpha\vartheta_L$ и перейти к шагу 4.

Шаг 6. Положить $x := x^t$, $J := J_t$ и перейти к шагу 2.

Критерий окончания процесса здесь не указан. Как правило, вычисления заканчиваются по исчерпанию заданного количества вычислений функционала либо при явной остановке алгоритма. Число пересчетов L по формуле (10) является параметром, задаваемым пользователем. Согласно (11) первоначально целесообразно полагать

$$L \cong \sqrt{1,63/\alpha_L} \cong 1,3\sqrt{\eta},$$

где η – оценка степени овражности минимизируемого функционала. При таком выборе L множители релаксации в положительной части спектра будут гарантированно меньше 0,23. При конструировании алгоритмических способов задания L необходимо учитывать, что последовательность $\{J_s\}$, где $J_s \triangleq J(x^k + \vartheta_s)$, не будет при $s \rightarrow \infty$ убывать монотонно. На шаге 5 алгоритма применена регулировка нормы вектора продвижения с целью предотвращения выхода из области справедливости локальной квадратичной модели функционала.

Характеристики сходимости

Дадим оценку эффективности метода (10) по сравнению с методами сопряженных градиентов (СГ-методами). Для задач

большой размерности (когда число итераций меньше размерности) можно гарантировать сходимость СГ-методов только со скоростью геометрической прогрессии даже для сильно выпуклых квадратичных функционалов.

Действительно, рассмотрим случай $f(x) = 1/2 \langle Gx, x \rangle$, $G > 0$ и оценим скорость сходимости метода СГ к экстремальной точке $x = 0$. Итерация x^k , полученная методом СГ, может быть представлена в виде [11]:

$$x^k = (E + c_1 G + c_2 G^2 + \dots + c_k G^k) x^0 = P_k(G) x^0,$$

где $P_k(G)$ – матричный полином k -й степени. При этом из свойств метода СГ следует, что коэффициенты c_1, \dots, c_k полинома $P_k(G)$ на каждой итерации принимают такие значения, чтобы минимизировать величину $f(x^k)$, только множителем отличающаяся от функции ошибки. Иначе говоря, k -е приближение минимизирует $f(x^k)$ среди векторов $x^0 + V$, где вектор V является элементом подпространства, натянутого на векторы $Gx^0, G^2x^0, \dots, G^kx^0$. Полагая

$$x^0 = \sum_{i=1}^n \xi_{i,0} u^i,$$

где $\{u^i\}$ – ортонормальный базис собственных векторов матрицы G , получим

$$x^k = P_k(G) \sum_{i=1}^n \xi_{i,0} u^i = \sum_{i=1}^n \xi_{i,0} P_k(\lambda_i) u^i, \quad P_k(0) = 1,$$

$$f(x^k) = 1/2 \langle Gx^k, x^k \rangle = 1/2 \sum_{i=1}^n \xi_{i,0}^2 P_k^2(\lambda_i) \lambda_i. \quad (13)$$

Отсюда имеем

$$\|x^0\|^2 = \sum_{i=1}^n \xi_{i,0}^2,$$

$$\|x^k\|^2 = \sum_{i=1}^n \xi_{i,0}^2 P_k^2(\lambda_i) \leq \max_i P_k^2(\lambda_i) \|x^0\|^2. \quad (14)$$

Выберем в качестве полинома $P_k(\lambda)$ близкий к оптимальному полином, наименее уклоняющийся от нуля на промежутке $[m, M]$, содержащем все собственные значения положительно определенной матрицы G , и нормированный так, что $P_k(0) = 1$.

Линейной заменой переменных

$$\lambda = \frac{M+m}{2} - \frac{M-m}{2}t$$

задача сводится к построению полинома, наименее отклоняющегося от нуля на промежутке $t \in [-1, 1]$ и принимающего в точке $t_0 = (M+m)/(M-m)$ (соответствующей $\lambda = 0$) значение 1. Решение последней задачи дается полиномом

$$\tilde{T}_k(t) = \frac{T_k(t)}{\cos(k \arccos t_0)} = \frac{T_k(t)}{T_k(t_0)},$$

где $T_k(t) = \cos(k \arccos t)$ – полином Чебышева. При этом

$$\max_{-1 \leq t \leq 1} |\tilde{T}_k(t)| = \frac{1}{|T_k(t_0)|} \max_{-1 \leq t \leq 1} |T_k(t)|.$$

Очевидно,

$$\max_{-1 \leq t \leq 1} |T_k(t)| = 1,$$

поэтому

$$\begin{aligned} L_k &= \max_{\lambda} |P_k(\lambda)| = \max_t |\tilde{T}_k(t)| = \\ &= \frac{1}{|T_k(t_0)|}, \lambda \in [m, M], t \in [-1, 1]. \end{aligned}$$

Так как справедливо представление

$$T_k(t) = 0,5 \left[(t + \sqrt{t^2 - 1})^k + (t - \sqrt{t^2 - 1})^k \right],$$

то

$$\begin{aligned} L_k &= 2 / \left[\left(\frac{M+m}{M-m} + \sqrt{\left(\frac{M+m}{M-m} \right)^2 - 1} \right)^k + \right. \\ &\quad \left. + \left(\frac{M+m}{M-m} - \sqrt{\left(\frac{M+m}{M-m} \right)^2 - 1} \right)^k \right] = \\ &= 2 / \left[\left(\frac{\sqrt{M} + \sqrt{m}}{\sqrt{M} - \sqrt{m}} \right)^k + \left(\frac{\sqrt{M} - \sqrt{m}}{\sqrt{M} + \sqrt{m}} \right)^k \right]. \end{aligned}$$

При достаточно больших k ($k \geq k_0$) имеем

$$\begin{aligned} L_k &\cong 2 / \left(\frac{\sqrt{M} + \sqrt{m}}{\sqrt{M} - \sqrt{m}} \right)^k = \\ &= 2 \left(\frac{\sqrt{\eta} - 1}{\sqrt{\eta} + 1} \right)^k \cong 2 \left(1 - \frac{2}{\sqrt{\eta}} \right)^k, \eta \triangleq \frac{M}{m}. \end{aligned} \quad (15)$$

Из (13) и (14) получаем

$$\|x^k\| \leq L_k \|x^0\|$$

или

$$\|x^k\| \leq 2q^k \|x^0\|, k \geq k_0, \quad (16)$$

где $q \cong (1 - 2/\sqrt{\eta})$. Таким образом, сходимость метода СГ со скоростью геометрической прогрессии доказана. Точное значение L_k , справедливое для любых k , будет при этом равно

$$L_k = 2 / \left[\left(1 + \frac{2}{\sqrt{\eta}} \right)^k + \left(1 - \frac{2}{\sqrt{\eta}} \right)^k \right], L_0 = 1.$$

Из (16) следует, что при $\eta \gg 1$ сходимость может быть очень медленной.

«Конечность» метода СГ, т. е. точное решение задачи минимизации квадратичной функции за n шагов, где n – размерность пространства поиска, проявляется только при достаточно большом количестве итераций. При этом степень полинома $P_k(\lambda)$ в (13) будет равна n , и оптимальный выбор этого полинома сводится к локализации его n корней в точках $\lambda_1, \lambda_2, \dots, \lambda_n$, что приведет к точному решению задачи ($f(x^n) = 0$).

Легко видеть, что оценка (16) для квадратичной функции общего вида

$$f(x) = 1/2 \langle Gx, x \rangle - \langle G, x \rangle + c$$

преобразуется к виду

$$\|x^k - x^*\| \leq 2q^k \|x^0 - x^*\|, \quad (17)$$

где x^* – оптимальная точка, не совпадающая в общем случае с началом координат.

Важная особенность алгоритмов типа RELCH заключается в том, что соответствующие множители релаксации будут определяться только числом итераций L и степенью η жесткости задачи независимо от размерности n . В то же время в схемах методов СГ для завершения каждого цикла спуска требуется порядка n итераций; в противном случае согласно (17) скорость сходимости может быть очень малой. Кроме того, каждая итерация метода СГ даже для квадратичного случая требует нового вычисления градиента, т. е. дополнительных вычислительных затрат по анализам функционирования оптимизируемой системы.

Будем далее полагать, что алгоритм RELCH реализован с постоянным $L = 1,3\sqrt{\eta}$, имея в области $\lambda > 0$ множители релаксации, не превышающие значения 0,23.

Рассмотрим задачу минимизации квадратичного функционала $f(x) = 1/2\langle Gx, x \rangle$ с положительно определенной матрицей G . Оценим количество вычислений $f(x)$, требуемое для достижения контрольного вектора x' с нормой $\|x'\| \leq 0,23$ методом СГ и алгоритмом RELCH из начальной точки x^0 с $\|x^0\| = 1$. По достижении точки x' вся ситуация повторяется, поэтому полученные ниже сравнительные оценки эффективности имеют достаточно общий характер.

Будем предполагать, что для вычисления производных применяются двусторонние конечноразностные соотношения, что в следующем ниже анализе дает дополнительные преимущества методу СГ.

Для достижения вектора x' алгоритму RELCH требуется вычислить в точке x^0 слабо заполненную матрицу Гессе и вектор градиента $f'(x^0)$. При коэффициенте заполнения γ это потребует около $2\gamma n^2$ вычислений f . Далее выполняется $L = 1,3\sqrt{\eta}$ итераций по формуле (10), не требующих дополнительных вычислений целевого функционала f .

Чтобы получить вектор x' методом СГ потребуется N итераций, где число N определяется из условия:

$$\|x^N\| = 2q^N = 0,23,$$

то есть $N \cong -2,2 / \ln q$. Для выполнения каждой итерации необходимо обновление вектора градиента, что связано с $2n$ вычислениями $f(x)$. Общее число вычислений f равно $-4,4n / \ln q$. Относительный выигрыш в количестве вычислений f методом RELCH по сравнению с методом СГ задается функцией $\Psi(\eta) \cong -2,2 / (\gamma n \ln q)$. Очевидно, при $\eta \rightarrow \infty$ имеем $q(\eta) \rightarrow 1$ и $\Psi(\eta) \rightarrow \infty$. Характерные значения Ψ для $\gamma = 0,01$ и $n = 1000$ даны ниже:

| | | | | | | |
|--------|-----|-----|------|------|--------|--------|
| η | ... | 100 | 1000 | 1500 | 10^4 | 10^5 |
| Ψ | ... | 1,0 | 3,4 | 4,0 | 11,0 | 35,0 |

Таким образом, для получения сравнимых результатов при $\eta = 10^4$ алгорит-

му RELCH потребуется приблизительно в 11 раз меньше вычислений f , чем методу СГ. Следует, однако, учитывать, что при увеличении η возрастает количество L пересчетов по формуле (10). Это может приводить к возрастанию влияния вычислительных погрешностей при вычислении ϑ_s , с большими номерами s .

Пример. Рассмотрим модельную задачу минимизации квадратичного функционала $f(x)$ с $n = 200$, $\eta = 1500$, $\gamma = 0,025$. Для определенности положим, что время однократного вычисления $f(x)$ эквивалентно выполнению 10^{2n} операций умножения с плавающей точкой. Время выполнения одной операции умножения для определенности и чисто условно положим равным $t_y = 3 \cdot 10^{-5}$ с. Вычисление значения $f(x)$ занимает при этом $t_f = 0,6$ с процессорного времени. Для вычисления f' и f'' с помощью общих конечноразностных формул требуется, соответственно, $t' = 2nt_f = 4$ мин, $t'' = 2\gamma n^2 t_y = 20$ мин. Число пересчетов по формуле (10) равно $L = 1,3\sqrt{\eta} = 50$. При каждом пересчете производится умножение слабо заполненной матрицы $E - 2G_k$ на вектор ϑ_s , что требует $\gamma n^2 t_y \cong 3 \cdot 10^{-2}$ с машинного времени. Время построения вектора ϑ_{50} без учета вычисления f', f'' составит около $50 \cdot 3 \cdot 10^{-2} = 1,5$ с и может в расчет не приниматься.

В результате получается, что для построения контрольного вектора x' с $\|x'\| < 0,23$ методом RELCH потребуется около $t'' = 20$ мин машинного времени. Метод СГ затратит, соответственно $\Psi(1500) \cdot 20 \cong 1,3$ ч.

При повторном применении алгоритма RELCH к построенному вектору x' мы получим вектор x'' с $\|x''\| \leq 0,23\|x'\|$ и т. д. Следовательно, если обозначить соответствующую последовательность векторов через $\{x^m\}$, то норма вектора x будет убывать по закону геометрической прогрессии $\|x^m\| \leq d^m \|x^0\|$, где $d < 0,23$ независимо от величины η и n .

Важным дополнительным преимуществом алгоритма RELCH по сравнению с методом СГ является его высокая эффективность в невыпуклом случае, т. к.

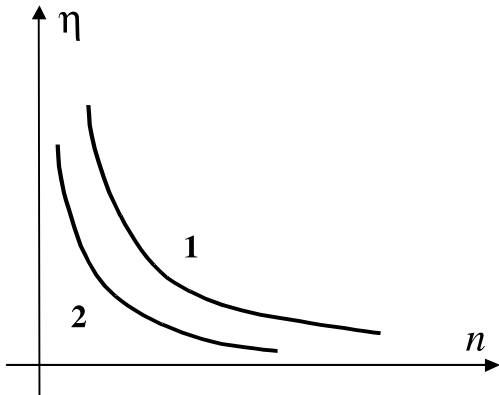


Рис. 2. Области работоспособности алгоритмов RELEX, RELCH $\varepsilon_M = \varepsilon'_M(1)$; $\varepsilon_M = \varepsilon''_M > \varepsilon'_M(2)$

функция релаксации метода в левой полуплоскости целиком расположена в разрешенной области и множители релаксации для $\lambda < 0$ быстро растут по абсолютной величине при переходе от ϑ_s к ϑ_{s+1} . Характеристики роста были приведены ранее.

Так же как и в случае ЭР-методов можно показать, что эффективность рассматриваемого подхода сохраняется при степенях жесткости, удовлетворяющих неравенству $\eta < 1/(n\varepsilon_M)$. Области работоспособности алгоритмов RELEX (ЭР-метод) и RELCH в плоскости (n, η) представлены на рис. 2. При умеренных размерностях n более эффективными, вообще говоря, оказываются алгоритмы типа RELEX. Они позволяют за меньшее число N_y операций умножения ма-

трицы на вектор получить заданные значения множителей релаксации. При больших η это приводит к существенному уменьшению накопленной вычислительной погрешности. Для подтверждения данного замечания достаточно проанализировать характер изменения множителей релаксации при применении формул пересчета при реализации ЭР-методов и (10). Характерные зависимости для рассмотренных случаев (для фиксированного $\lambda_i < 0$) и разных ε_M представлены на рис. 3.

Видно, что если область локальной квадратичности функционала $J(x)$ невелика (ζ_k мало), то необходимые значения $|R_i| \cong 1$ и более эффективными могут оказаться методы типа RELCH.

Описанный класс матричных градиентных методов показал на практике достаточно эффективную работу в условиях высокой жесткости и невыпуклости целевых функционалов. При оптимизации больших систем удается использовать эффективные «упакованные» формы хранения матриц вторых производных, что существенно снижает требования к необходимой компьютерной памяти. Однако метод сохраняет свои основные характеристики и для малоразмерных систем, выдерживая конкуренцию с основными оптимизационными процедурами нелинейного программирования.

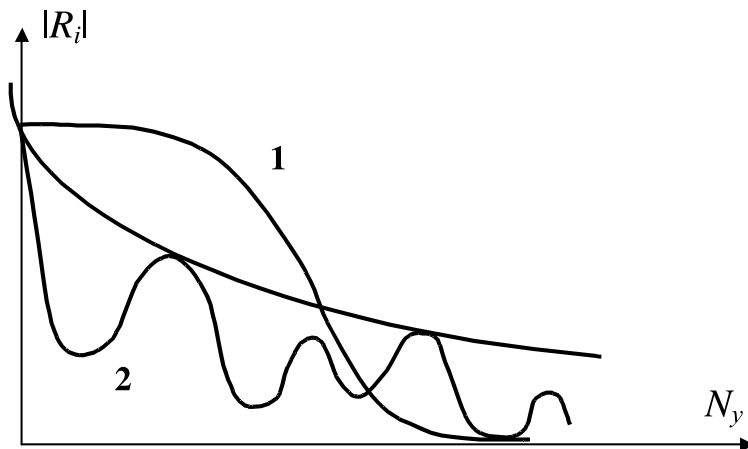


Рис. 3. Характер изменения множителей релаксации: 1 – RELEX, 2 – RELCH

СПИСОК ЛИТЕРАТУРЫ

1. **Черноруцкий И.Г.** Методы оптимизации. Компьютерные технологии. СПб.: БХВ-Петербург, 2011. 384 с.
2. **Черноруцкий И.Г.** Методы параметрической оптимизации в задачах идентификации // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. СПб.: Изд-во СПбГПУ, 2009. № 2(76). С. 150–155.
3. **Черноруцкий И.Г.** Параметрические методы синтеза систем управления // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. СПб.: Изд-во СПбГПУ, 2009. № 2(76). С. 111–115.
4. **Черноруцкий И.Г.** Алгоритмические проблемы жесткой оптимизации // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. СПб.: Изд-во СПбГПУ, 2012. № 6(162). С. 141–152.
5. **Ракитский Ю.В., Устинов С.М., Черноруцкий И.Г.** Численные методы решения жестких систем. М.: Наука, 1979. 208 с.
6. **Черноруцкий И.Г.** Функции релаксации градиентных методов // Научно-технические ведомости СПбГПУ. Информатика. Телекомму-
- никации. Управление. СПб.: Изд-во СПбГПУ, 2012. № 3(150). С. 66–72.
7. **Черноруцкий И.Г.** Некоторые стандартные схемы параметрической оптимизации // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. СПб.: Изд-во СПбГПУ, 2012. № 6(162). С. 128–133.
8. **Черноруцкий И.Г.** Градиентные методы с экспоненциальной функцией релаксации // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. СПб.: Изд-во СПбГПУ, 2013. № 5 (181). С. 58–66.
9. **Брамеллер А., Аллан Р., Хэмэм Я.** Слабозаполненные матрицы. М.: Энергия, 1979.
10. **Джордж А., Лю Дж.** Численное решение больших разреженных систем уравнений. М.: Мир, 1984.
11. **Фаддеев Д.К., Фаддеева В.Н.** Вычислительные методы линейной алгебры. М.: Физматгиз, 1963.
12. **Ланцош К.** Практические методы прикладного анализа. М.: Физматгиз, 1961.

REFERENCES

1. **Chernorutskiy I.G.** *Metody optimizatsii. Kompyuternyye tekhnologii.* St. Petersburg: BKhV-Petersburg Publ., 2011, 384 p. (rus)
2. **Chernorutskiy I.G.** *Metody parametricheskoy optimizatsii v zadachakh identifikatsii* [Methods of parametrical optimisation in identification problems], *Nauchno-tekhnicheskkiye vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravlenie.* St. Petersburg: SPbGPU Publ., 2009, No. 2(76), Pp. 150–155. (rus)
3. **Chernorutskiy I.G.** *Parametricheskkiye metody sinteza sistem upravleniya* [Parametrical methods of synthesis of control systems], *Nauchno-tekhnicheskkiye vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravlenie.* St. Petersburg: SPbGPU Publ., 2009, No. 2(76), Pp. 111–115. (rus)
4. **Chernorutskiy I.G.** *Algoritmicheskkiye problemy zhestkoy optimizatsii* [Algorithmic problems of stiff optimization], *Nauchno-tekhnicheskkiye vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravlenie.* St. Petersburg: SPbGPU Publ., 2012, No. 6(162), Pp. 141–152. (rus)
5. **Rakitskiy Yu.V., Ustinov S.M., Chernorutskiy I.G.** *Chislennyye metody resheniya zhestkikh sistem.* Moscow: Nauka Publ., 1979, 208 p. (rus)
6. **Chernorutskiy I.G.** *Funktsii relaksatsii* *gradiyentnykh metodov* [Relaxation function of gradient methods], *Nauchno-tekhnicheskkiye vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravlenie.* St. Petersburg: SPbGPU Publ., 2012, No. 3(150), Pp. 66–72. (rus)
7. **Chernorutskiy I.G.** *Nekotoryye standartnyye skhemy parametricheskoy optimizatsii* [Standard schemes of parametric optimization], *Nauchno-tekhnicheskkiye vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravlenie.* St. Petersburg: SPbGPU Publ., 2012, No. 6(162), Pp. 128–133. (rus)
8. **Chernorutskiy I.G.** *Gradiyentnyye metody s eksponentsialnoy funktsiyey relaksatsii* [Gradient methods with exponent relaxation function], *Nauchno-tekhnicheskkiye vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravlenie.* St. Petersburg: SPbGPU Publ., 2013, No. 5 (181), Pp. 58–66. (rus)
9. **Brameller A., Allan R., Khemem Ya.** *Slabozapolnennyye matritsy.* Moscow: Energiya Publ., 1979. (rus)
10. **Dzhordzh A., Lyu Dzh.** *Chislennoye resheniye bolshikh razrezhennykh sistem uravneniy.* Moscow: Mir Publ., 1984. (rus)
11. **Faddeyev D.K., Faddeyeva V.N.** *Vychislitelnyye*

metody lineynoy algebry. Moscow: Fizmatgiz Publ., 1963. (rus)

12. **Lantsosh K.** *Prakticheskiye metody prikladnogo analiza*. Moscow: Fizmatgiz Publ., 1961. (rus)

ЧЕРНОРУЦКИЙ Игорь Георгиевич – *заведующий кафедрой информационных и управляющих систем Института информационных технологий и управления Санкт-Петербургского государственного политехнического университета, доктор технических наук.*

195251, Россия, Санкт-Петербург, ул. Политехническая, д. 29.

E-mail: igcher1946@mail.ru

CHERNORUTSKIY, Igor G. *St. Petersburg State Polytechnical University.*

195251, Politekhnikeskaya Str. 29, St. Petersburg, Russia.

E-mail: igcher1946@mail.ru