

СТЕНИН И. В., ШАМАНАЕВ П. А.

АЛГОРИТМ РЕШЕНИЯ РАЗРЕЖЕННОЙ СИСТЕМЫ ЛИНЕЙНЫХ  
АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ БОЛЬШОЙ РАЗМЕРНОСТИ  
С ИСПОЛЬЗОВАНИЕМ МЕТОДА СОПРЯЖЕННЫХ ГРАДИЕНТОВ

**Аннотация.** В статье описан алгоритм решения разреженных систем линейных алгебраических уравнений большой размерности специального вида с использованием метода сопряженных градиентов. Приведены результаты вычислительного эксперимента при различных значениях входных данных.

**Ключевые слова:** система линейных алгебраических уравнений большой размерности, разреженная матрица, метод сопряженных градиентов, пространства Крылова, биортогонализация Ланцоша.

STENIN I. V., SHAMANAEV P. A.

ALGORITHM OF SOLVING A SPARSE SYSTEM OF LINEAR ALGEBRAIC  
EQUATIONS OF LARGE DIMENSION BY CONJUGATE GRADIENT METHOD

**Abstract.** The algorithm of solving sparse systems of linear algebraic equations of large dimension of a special type by using of the conjugate gradient method is described. The results of a computational experiment for different values of the input data are presented.

**Keywords:** system of linear algebraic equations of large dimension, sparse matrix, conjugate gradient method, Krylov spaces, Lanczos biorthogonalization.

При решении задач идентификации параметров динамических систем по экспериментальным данным [1–3] возникает необходимость решения разреженных систем линейных алгебраических уравнений большой размерности специального вида.

Рассмотрим систему линейных алгебраических уравнений следующего вида:

$$My = d, \quad (1)$$

где

$$M = \begin{pmatrix} H & T^T & A^T \\ T & O & O \\ A & O & O \end{pmatrix}, \quad y = \begin{pmatrix} s \\ \mu \\ \lambda \end{pmatrix}, \quad d = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix},$$

$s, d_1 \in R^{2N+4}$ ,  $\mu, d_2 \in R^2$ ,  $\lambda, d_3 \in R^{2(N-1)}$ ,  $O$  – нулевые матрицы соответствующих размеров,

$$H = \begin{pmatrix} & & & & & B_{1,\theta} \\ & & & & & B_{2,\theta} \\ & & \frac{1}{N} I_{2N} & & & \vdots \\ & & & & & B_{N-1,\theta} \\ & & & & & B_{N,\theta} \\ B_{\theta,1} & B_{\theta,2} & \dots & B_{\theta,N-1} & B_{\theta,N} & B_{\theta,\theta} \end{pmatrix}, \quad \begin{aligned} B_{\theta,i} &= B_{i,\theta}^T, \\ B_{i,\theta} &= O_{2 \times 4}, \\ i &= \overline{1, N}, \\ B_{\theta,\theta} &= O_4. \end{aligned}$$

$I_{2N}$  — единичная  $(2N \times 2N)$ -матрица,  $O_4$  — нулевая  $(4 \times 4)$ -матрица,  $O_{2N \times 4}$  — нулевая  $(2N \times (2N + 4))$ -матрица,

$$T = [T_1, T_2, \dots, T_N, T_\theta],$$

$T_1$  — единичная  $(2 \times 2)$ -матрица,  $T_i$  ( $i = \overline{2, N}$ ) — нулевые  $(2 \times 2)$ -матрицы,  $T_\theta$  — нулевая  $(2 \times 4)$ -матрица,

$$A = \begin{pmatrix} D_1 & K_1 & & & E_1 \\ & D_2 & K_2 & & E_2 \\ & & \ddots & \ddots & \\ & & & D_{N-1} & K_{N-1} & E_{N-1} \end{pmatrix},$$

$$D_i = \begin{pmatrix} 1 + \frac{\tau}{2} \theta_1 & \frac{\tau}{2} \theta_2 \\ \frac{\tau}{2} \theta_3 & 1 + \frac{\tau}{2} \theta_4 \end{pmatrix}, \quad K_i = \begin{pmatrix} -\left(1 - \frac{\tau}{2} \theta_1\right) & \frac{\tau}{2} \theta_2 \\ \frac{\tau}{2} \theta_3 & -\left(1 - \frac{\tau}{2} \theta_4\right) \end{pmatrix}, \quad \theta_k \in \mathbb{R}, \quad k = \overline{1, 4},$$

$$E_i = \begin{pmatrix} \frac{\tau}{2}(x_{i,1} + x_{i+1,1}) & \frac{\tau}{2}(x_{i,2} + x_{i+1,2}) & 0 & 0 \\ 0 & 0 & \frac{\tau}{2}(x_{i,1} + x_{i+1,1}) & \frac{\tau}{2}(x_{i,2} + x_{i+1,2}) \end{pmatrix}, \quad i = \overline{1, N-1}.$$

Компоненты вектора  $d$  вычисляются по формулам

$$d_1 = -H_2(z - \tilde{z}), \quad d_2 = h - Tz, \quad d_3 = -g(z),$$

$$g(z) = \text{column}(g_1(z), \dots, g_{N-1}(z)), \quad g_i(z) = \text{column}(g_{i,1}(z), g_{i,2}(z)), \quad i = \overline{1, N-1}.$$

$$z = \text{column}(\mathbf{x}_N, \theta) \in \mathbb{R}^{2N+4}, \quad \tilde{z} = \text{column}(\mathbf{x}_N, O_4) \in \mathbb{R}^{2N+4}, \quad H_2 = \frac{1}{N} H_1^T H_1,$$

$$\mathbf{x}_N = \text{column}(x_1, \dots, x_N) \in \mathbb{R}^{2N}, \quad x_i = \text{column}(x_{i,1}, x_{i,2}) \in \mathbb{R}^2, \quad i = \overline{1, N},$$

$$g_{i,1}(z) \equiv g_{i,1}(\mathbf{x}_N, \theta) = \left(1 + \frac{\tau}{2} \theta_1\right) x_{i,1} + \frac{\tau}{2} \theta_2 x_{i,2} - \left(1 - \frac{\tau}{2} \theta_1\right) x_{i+1,1} + \frac{\tau}{2} \theta_2 x_{i+1,2}, \quad i = \overline{1, N-1},$$

$$g_{i,2}(z) \equiv g_{i,2}(\mathbf{x}_N, \theta) = \frac{\tau}{2} \theta_3 x_{i,1} + \left(1 + \frac{\tau}{2} \theta_4\right) x_{i,2} + \frac{\tau}{2} \theta_3 x_{i+1,1} - \left(1 - \frac{\tau}{2} \theta_4\right) x_{i+1,2}, \quad i = \overline{1, N-1},$$

Для приближенного нахождения решения системы (1) использовался метод сопряженных градиентов. Приведем алгоритм метода сопряженных градиентов [4] для системы (1):

1. Выберем начальное приближение  $y^{(0)}$ .
2.  $r^{(0)} := d - My^{(0)}$ ,  $p^{(0)} := r^{(0)}$ ,  $j := 1$ .
3. Вычислим коэффициент  $\alpha_j := (r^{(j)}, r^{(j)}) / (Mr^{(j)}, p^{(j)})$ .
4. Найдем следующее приближение  $y^{(j+1)} := y^{(j)} + \alpha_j p^{(j)}$ .
5. Вычислим поправку к решению  $r^{(j+1)} := r^{(j)} - \alpha_j Mr^{(j)}$ .
6. Найдем коэффициент  $\beta_j := (r^{(j+1)}, r^{(j+1)}) / (r^{(j)}, r^{(j)})$ .
7. Если  $\|r^{(j+1)}\| < \varepsilon$ , то алгоритм завершается.
8. Вычислим вектор, вдоль которого вычисляется поправка  $p^{(j+1)} := r^{(j+1)} + \beta_j p^{(j)}$ .
9.  $j := j + 1$ .
10. Перейдем к пункту 3.

На основании приведенного алгоритма разработано программное обеспечение на языке C#. Структура программного обеспечения имеет вид

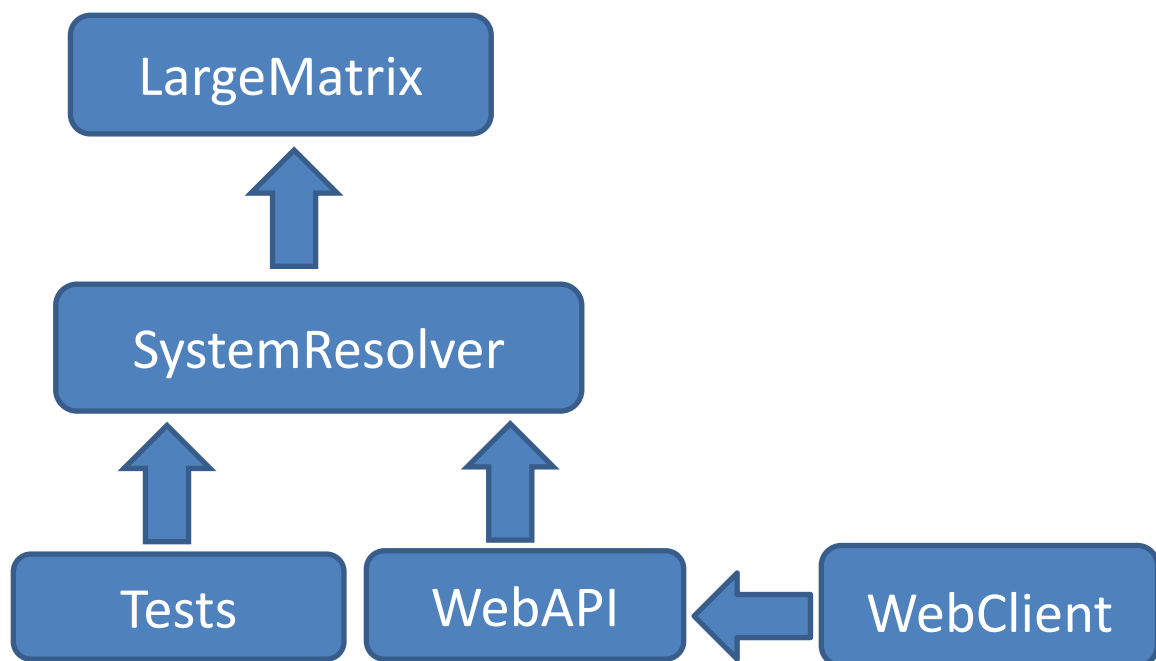


Рис. 1. Структура программного обеспечения.

Опишем модули программного обеспечения.

**LargeMatrix** – библиотека, содержащая классы разреженной матрицы, плотного вектора и плотной матрицы, а также классов, отвечающих за построение системы уравнений заданного вида.

**Tests** – модульные тесты, покрывающие библиотеку LargeMatrix, а также метод решения СЛАУ.

**SystemResolver** – основной проект, содержащий алгоритм решения разреженной системы методом сопряженных градиентов.

**WebAPI** – API на ASP.NET, позволяющий делать запрос к серверу на получение данных решения.

**WebClient** – клиентское приложение, получающее и отображающее данные решения в браузере.

Для хранения разреженной матрицы использовался разреженный строчный формат (Compressed sparse row), который предполагает наличие трех одномерных массивов.

- Массив, содержащий все ненулевые элементы матрицы.
- Массив, содержащий номера столбцов для соответствующих элементов.
- Массив индексов элементов, с которых начинается описание строки.

Вычислительный эксперимент проводился при следующих начальных данных:

$$N = 10, \theta_1 = -1, \theta_2 = 1, \theta_3 = -0.5, \theta_4 = 0.5, h = \text{column}(100, 150).$$

Значения  $x_{i,1}$  и  $x_{i,2}$  ( $i = \overline{1, 10}$ ) приведены в таблице 1.

Таблица 1

Значения  $x_{i,1}$  и  $x_{i,2}$ ,  $i = \overline{1, 10}$

$i$	1	2	3	4	5	6	7	8	9	10
$x_{i,1}$	110	190	200	333	355	383	400	415	420	445
$x_{i,2}$	151	163	175	204	222	244	266	333	355	380

Результаты вычислительного эксперимента при различных значениях погрешности вычислений  $\varepsilon$  и параметра  $\tau$  приведены в таблице 2.

Количество итераций и время вычислений при различных  $\varepsilon$  и  $\tau$ 

Погрешность, $\varepsilon$	$\tau = 0.5$	$\tau = 0.1$	$\tau = 0.05$	$\tau = 0.01$
0.1	32 0,003185 сек.	27 0,003003 сек.	25 0,002839 сек.	24 0,003031 сек.
0.01	33 0,003587 сек.	27 0,003042 сек.	25 0,003219 сек.	24 0,003095 сек.
0.001	33 0,004341 сек.	30 0,003109 сек.	30 0,003307 сек.	24 0,003372 сек.

Анализ таблицы 2 показывает, что при уменьшении параметра  $\tau$  количество итераций, требуемых для достижения заданной точности  $\varepsilon$ , уменьшается, и, следовательно, скорость сходимости итерационного метода увеличивается. Вместе с тем, при фиксированном значении параметра  $\tau$  и уменьшении погрешности вычислений  $\varepsilon$  время вычислений и количество итераций увеличивается незначительно.

#### ЛИТЕРАТУРА

1. Атряхин В. А., Чельшов М. С., Шаманаев П. А. Применение метода ортогональной циклической редукции для решения систем линейных алгебраических уравнений с матрицами специального вида [Электронный ресурс]// Огарев-online. Раздел "Физико-математические науки". – 2014. – № 19. – Режим доступа: <http://journal.mrsu.ru/arts/primenenie-metoda-ortogonalnojj-ciklicheskojj-redukcii-dlya-resheniya-sistem-linejjnykh-algebraicheskikh-uravnenijj-s-matricami-specialnogo-vida>.
2. Чельшов М. С., Шаманаев П. А. Идентификация параметров динамических систем на основе экспериментальных данных // Актуальные вопросы прикладной математики и информатики: сб. науч. тр. – Саранск: СВМО, 2015. – С. 39–42.
3. Zhengfeng Li, Michael R. Osborne, Tania Prvan Parameter estimation of ordinary differential equations // IMA Journal of Numerical Analysis. – 2005. – No. 25. – pp. 264–285.
4. Баландин М. Ю., Шурина Э. П. Методы решения СЛАУ большой размерности. – Новосибирск: НГТУ, 2000. – 70 с.