УДК 519.71
ББК 22.193

ИТЕРАЦИОННЫЙ МЕТОД РЕШЕНИЯ СИСТЕМ МАТРИЧНЫХ УРАВНЕНИЙ

Зюзина Н.Ю.[1]
(*Арзамасский политехнический институт, Арзамас*)
Аминов И.Л.[2]
(*Санкт-Петербургский государственный университет аэрокосмического приборостроения, Санкт-Петербург*)

В данной статье проанализировано итерационное решение систем матричных уравнений Сильвестра. Разработан и реализован рекурсивный алгоритм решения систем матричных уравнений.

Ключевые слова: матричные уравнения Сильвестра, итерационные методы, рекурсивный алгоритм решения.

1. Введение

Теория матриц и матричные уравнения давно вошли в число основных инструментов, используемых различными математическими дисциплинами. В то же время они сами являются плодотворной областью исследований. Матричные уравнения необходимы практически в любой области математики – дифференциальные уравнения, теория вероятности и статистика, теория оптимизации – и практически во всех ее приложениях – приложения к теоретической и прикладной экономике, инженерным дисциплинам или исследованию операций [1].

[1] *Зюзина Наиля Юрьевна, старший преподаватель* (*emelyanova@apingtu.edu.ru*).
[2] *Аминов Иван Леонидович, студент* (*emelyanova@apingtu.edu.ru*).

До недавнего времени необходимая информация по применению и решению матричных уравнений появлялась редко или не появлялась вообще. По мере того как рос интерес к прикладной математике и теории матриц, этим темам посвящалось все больше литературы и матричные уравнения находили свое применения во многих областях математики.

Существует большое число итерационных методов для решения матричных уравнений, например, такие как методы Якоби и Гаусса-Зейделя.

В данной работе рассмотрен итерационный метод решения систем матричных уравнений Сильвестра, которые применяются во многих системах управления, а также в анализе устойчивости. Метод реализован в системе компьютерной математики MATLAB.

2. *Итерационные методы решения систем матричных уравнений*

Рассмотрим линейное уравнение вида:

(1) $Ax = b, \quad A \in R^{n \times n}, \quad b \in R^n$,

где $A=|a_{ij}|$ – это полноранговая матрица с ненулевыми диагональными элементами и $x \in R^n$ – неизвестный вектор.

Пусть $G_k \in R^{nxn}$ – матрица, которую необходимо определить, и пусть $\mu>0$– длина шага или множитель сходимости. Большое семейство итерационных методов можно представить следующим образом [2]:

(2) $x(k) = x(k-1) + \mu G_k [b - Ax(k-1)], \quad k = 1, 2, 3, \ldots$.

Теорема 1. Для итерационного алгоритма (2), предположим, что система (1) имеет единственное решение. Тогда итерационное решение $x(k)$ в выражении (2) сходится к точному решению x (т.е. $\lim\limits_{x \to \infty} x(k) = x$) при любых начальных значениях $x(0)$, если существует $\varepsilon>0$, независящее от k, такое, что выполняется условие [2]:

(3) $\mu(G_k A)^T (G_k A) + \varepsilon I \le (G_k A)^T + (G_k A)$ для всех k.

Если матрица $(G_kA)^T+(G_kA)$ положительно определена, то множитель сходимости можно получить из следующего соотношения [2]:

$$0 < \mu < \frac{\lambda_{\min}\left[(G_k A)^T + (G_k A)\right]}{\lambda_{\max}\left[(G_k A)^T (G_k A)\right]} \quad \text{для всех } k,$$

где $\lambda_{\max}(\lambda_{\max})$ – максимальное (минимальное) собственное значение.

Следствие. Если $G_k=A^T$, тогда градиентный итерационный алгоритм [2]:

$$(4) \quad \begin{cases} x(k) = x(k-1) + \mu A^T\left[b - Ax(k-1)\right], \quad k = 1, 2, 3, \ldots, \\ 0 < \mu < \dfrac{2}{\lambda_{\max}[A^T A]} \; \text{или} \; 0 < \mu < \dfrac{2}{\|A\|^2}, \end{cases}$$

приводит к тому, что $\lim\limits_{x\to\infty} x(k) = x$. Здесь $\|X\|^2 = tr\left[XX^T\right]$.

Применим иерархический принцип идентификации для решения системы матричных уравнений Сильвестра:

$$(5) \quad \begin{cases} AX + YB = C, \\ DX + YE = F. \end{cases}$$

где A, $D \in \mathrm{R}^{mxm}$, B, $E \in \mathrm{R}^{nxn}$ и C, $F \in \mathrm{R}^{mxn}$ заданные постоянные матрицы; X, $Y \in \mathrm{R}^{mxn}$ – неизвестные матрицы.

Введем обозначение: для двух матриц X и Y размера mxn:

$$X = \left[x_1, x_2, \ldots, x_n\right] \in R^{m\times n},$$

$$col\left[X\right] = \left[x_1 \;\; x_2 \ldots x_n\right]^T,$$

$$col\left[X, Y\right] = \begin{bmatrix} col[X] \\ col[Y] \end{bmatrix} \in R^{2m\times n}.$$

Лемма. Система (5) имеет единственное решение [2]:

$$(6) \quad col\left[X, Y\right] = S_2^{-1} col\left[C, F\right],$$

тогда и только тогда, когда матрица

$$S_2 := \begin{bmatrix} I_n \otimes A & B^T \otimes I_m \\ I_n \otimes D & E^T \otimes I_m \end{bmatrix} \in R^{(2mn)\times(2mn)}$$

является невырожденной и соответствующее однородное матричное уравнение ($AX+YB=0$, $DX+YE=0$) имеет единственное решение: $X=Y=0$.

Согласно иерархическому принципу идентификации, система (5) разбивается на две подсистемы и затем, основываясь на принципе поиска градиента, определяются параметры каждой подсистемы [2].

Определим 2 матрицы:

(7) $$b_1 := \begin{bmatrix} C - YB \\ F - YE \end{bmatrix},$$

(8) $$b_2 := \left[C - AX, \;\; F - DX \right].$$

Затем из (5) получаем две подсистемы:

$$S_1: \;\; \begin{bmatrix} \mathrm{A} \\ \mathrm{D} \end{bmatrix} X = b_1,$$

$$S_2: \;\; Y\left[B, \; E\right] = b_2.$$

Пусть $X(k)$ и $Y(k)$ – приближенные или итерационные решения X и Y, связанные с подсистемами S_1 и S_2. Используя принцип поиска градиента, приходим к следующим рекурсивным уравнениям [2]:

(9) $$X(k) = X(k-1) + \mu \begin{bmatrix} A \\ D \end{bmatrix}^T \left\{ b_1 - \begin{bmatrix} A \\ D \end{bmatrix} X(k-1) \right\},$$

(10) $$Y(k) = Y(k-1) + \mu \{ b_2 - Y(k-1)[B, \;\; E] \} [B, \;\; E]^T.$$

Здесь μ>0– это величина шага итерации или множитель сходимости, который определяется ниже. Подставляя (7) в (9) и (8) в (10) получим [2]:

(11) $$X(k) = X(k-1) + \mu \begin{bmatrix} A \\ D \end{bmatrix}^T \left\{ \begin{bmatrix} C - YB \\ F - YE \end{bmatrix} - \begin{bmatrix} A \\ D \end{bmatrix} X(k-1) \right\} =$$

$$= X(k-1) + \mu \begin{bmatrix} A \\ D \end{bmatrix}^T \begin{bmatrix} C - YB - AX(k-1) \\ F - YE - DX(k-1) \end{bmatrix},$$

(12) $$Y(k) = Y(k-1) + \mu \{ [C - AX, \;\; F - DX] - Y(k-1)[B, \;\; E] \} [B, \;\; E]^T =$$

$$= Y(k-1) + \mu [C - AX - Y(k-1)B, \;\; F - DX - Y(k-1)E][B, \;\; E]^T.$$

Поскольку выражение в правой части (11) и (12) содержит неизвестные матрицы-параметры X и Y, невозможно использовать алгоритм в (11) и (12). Согласно иерархическому принципу идентификации, неизвестные переменные X в (11) и Y в (12) заменяются их оценками $X(k\text{-}1)$ и $Y(k\text{-}1)$ в момент $(k\text{-}1)$. Отсюда получаем итерационные решения $X(k)$ и $X(k)$ для системы уравнений Сильвестра (5) [2]:

(13) $$X(k) = X(k-1) + \mu \begin{bmatrix} A \\ D \end{bmatrix}^T \begin{bmatrix} C - AX(k-1) - Y(k-1)B \\ F - DX(k-1) - Y(k-1)E \end{bmatrix},$$

(14) $Y(k) = Y(k-1) +$
$+ \mu[C - AX(k-1) - Y(k-1)B,\ \ F - DX(k-1) - Y(k-1)E][B,\ \ E]^T.$

Множитель сходимости может быть выбран из условия [2]:

(15) $$0 < \mu < \frac{2}{\|A\|^2 + \|B\|^2 + \|D\|^2 + \|E\|^2}.$$

Чтобы инициализировать алгоритм, примем $X(0)=Y(0)=0$ или вещественную матрицу, например, $X(0)=Y(0)=10^{-6}\mathbf{1}_{mxn}$**,** где $\mathbf{1}_{mxn}$ – матрица mxn, элементы которой равны единице.

Теорема 2. Если система уравнений Сильвестра (5) имеет единственное решение X и Y, то для любых начальных значений итерационные решения $X(k)$ и $Y(k)$ получаемые с помощью алгоритма (13)–(14), сходятся к решениям X и Y [2]:

$$\lim_{k\to\infty} X(k) = X, \qquad \lim_{k\to\infty} Y(k) = Y.$$

Для того чтобы улучшить сходимость, обычно выбирают большой множитель сходимости μ, который приводит к высокой скорости сходимости $X(k)$ к X, $Y(k)$ к Y; но такое большое μ может нарушать условие этой теоремы. Обычно существует некоторое лучшее μ такое, что достигается высокая скорость сходимости.

3. Программная реализация

Для решения системы матричных уравнений Сильвестра был разработан алгоритм, основанный на рекурсивных формулах (13)-(14).

Основная идея алгоритма состоит в том, чтобы рассматривать неизвестные матрицы $X(k\text{-}1)$ и $Y(k\text{-}1)$ как параметры системы, и получить итерационные решения $X(k)$, $Y(k)$, применяя иерархический принцип идентификации. Для наглядной демонстрации сходимости алгоритма необходимо построить график относительной итерационной ошибки $\delta(k)$ при разных значениях параметра μ.

Рассмотрим пример, иллюстрирующий работу описанного алгоритма. Предположим, что система матричных уравнений Сильвестра представлена в следующем виде:

$AX + YB = C$ и $DX + YE = F$,

где

$$A = \begin{bmatrix} 2.00 & 1.00 \\ -1.00 & 2.00 \end{bmatrix}, \quad B = \begin{bmatrix} 1.00 & -0.20 \\ 0.20 & 1.00 \end{bmatrix}, \quad C = \begin{bmatrix} 13.20 & 10.60 \\ 0.60 & 8.40 \end{bmatrix},$$

$$D = \begin{bmatrix} -2.00 & -0.50 \\ 0.50 & 2.00 \end{bmatrix}, \quad E = \begin{bmatrix} -1.00 & -3.00 \\ 2.00 & -4.00 \end{bmatrix}, \quad F = \begin{bmatrix} -9.50 & -18.00 \\ 16.00 & 3.50 \end{bmatrix}.$$

Тогда единственными решениями X и Y из (6) являются:

$$X = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix} = \begin{bmatrix} 4.00 & 3.00 \\ 3.00 & 4.00 \end{bmatrix}, \quad Y = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix} = \begin{bmatrix} 2.00 & 1.00 \\ -2.00 & 3.00 \end{bmatrix}.$$

В качестве начальных значений возьмем $X(0)=Y(0)=10^{-6}\mathbf{1}_{2x2}$,
$0 < \mu < 2 \Big/ \left[\|A\|^2 + \|B\|^2 + \|D\|^2 + \|E\|^2 \right] = 2/38.47$.

Результаты работы программы, а именно итерационные решения $X(k)$ и $Y(k)$ показаны в таблицах 1, 2, а характер изменения относительной итерационной ошибки δ, рассчитанной по формуле

$$\delta = \sqrt{\frac{\|X(k) - X\|^2 + \|Y(k) - Y\|^2}{\|X\|^2 + \|Y\|^2}}$$

и представлен на рисунке 1 и в таблице 3.

Таблица 1. Итерационные решения X(k)

k	x_{11}	x_{12}	x_{21}	x_{22}
1	2.9333	2.8083	2.8417	2.4111
…	…	…	…	…
36	3.9921	2.9622	3.0046	3.9821
…	…	…	…	…
308	4.0000	3.0000	3.0000	4.0000

Таблица 2. Итерационные решения Y(k)

k	y_{11}	y_{12}	y_{21}	y_{22}
1	4.1433	3.6800	-1.5322	1.4733
…	…	…	…	…
36	2.0015	1.0019	-2.0025	2.9997
…	…	…	…	…
308	2.0000	1.0000	-2.0000	3.0000

Таблица 3. Итерационная ошибка δ

k	μ=1/101.16	μ=1/38.47	μ=1/30	μ=1/20
1	0.8840	0.7154	0.6521	0.5604
…	…	…	…	…
36	0.5742	0.5723	0.5719	0.5715
…	…	…	…	…
308	0.5714	0	0	0

Из таблицы 3 видно, что при увеличении k ошибка уменьшается и приближается к 0. Это доказывает эффективность алгоритма.

Эффект изменения множителя сходимости μ показан на рисунке 1. Уменьшение множителя сходимости μ приводит к увеличению скорости сходимости алгоритма, но при этом возрастает итерационная ошибка (табл.3). Если увеличить μ до 1/20, то наблюдается неустойчивая сходимость алгоритма (рис.1). Таким образом, был определено оптимальное значение множителя сходимости μ=1/38.47, при котором достигается максимальная скорость сходимости алгоритма.

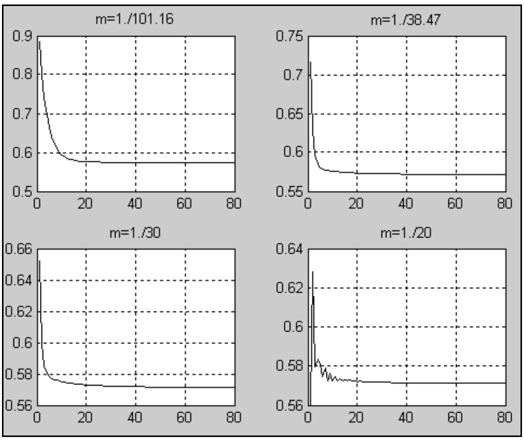

Рис.1. График изменения относительной итерационной ошибки δ для системы матричных уравнений Сильвестра.

4. Заключение

В результате выполнения данной работы были решены следующие задачи:

- проанализирован вывод рекуррентных соотношений для решения матричных уравнений;
- исследуемые алгоритмы были реализованы в системе программирования MATLAB;
- проиллюстрирована эффективность алгоритма на примере.

Основным достоинством рассмотренных алгоритмов является использование меньшего количества памяти, чем существующие численные методы. А их реализация в системе MATLAB позволила значительно упростить и сократить код программы.

Итерационный метод, предложенный для решения систем матричных уравнений, может быть расширен для решения других линейных или нелинейных матричных уравнений, например, уравнения Риккати.

Литература

1. ХОРН Р., ДЖОНСОН Ч. *Матричный анализ*. – М.: Мир, 1989. – 665 с.
2. DING, F., CHEN, T. On iterative solutions of general coupled matrix equations. // SIAM J. Control Optim. – 2006. – Vol. 44, № 6. – P. 2269-2284.

ITERATIVE METHODS FOR SOLVING MATRIX EQUATIONS SYSTEMS

Nailya Zuzina, Arzamas Polytechnic Institute, Arzamas, senior teacher (*emelyanova@apingtu.edu.ru*).
Ivan Aminov, St. Petersburg State University of Aerospace Instrumentation, St. Petersburg, student (*emelyanova@apingtu.edu.ru*).

Abstract: This article analyzed the iterative solution of systems of matrix equations Sylvester. Developed and implemented a recursive algorithm for solving systems of matrix equations.

Keywords: Sylvester matrix equations, iterative Methods, recursive algorithm for solving.