Physics and Mathematics

UDC 517.97, 519.612

Tsukanova Alisa

Candidate of Physics and Mathematics,

Assistant of the Department of Mathematical Physics and Differential Equations

National Technical University of Ukraine

«Igor Sikorsky Kiev Polytechnic Institute»

HOW TO APPLY VARIATIONAL METHODS TO CLASSICAL PROBLEMS FROM ALGEBRA

Summary. *The given paper presents basic results of analysis of well-known classical Gaussian elimination method of successive elimination of unknowns in solving some set of linear equations and «modern» optimization methods, realized via programming. It is shown that some mathematical problems can be solved with the help of optimization apparatus. Computer-based methods are ideally suited to its implementation.*

Key words: *linear algebra, systems of linear algebraic equations, Gaussian elimination method, Kronecker-Capelli theorem, variational methods, optimization methods, gradient descent method, gradient, anti-gradient, minimizer, «Visual Basic for Applications».*

Introduction. Optimization in mathematical sense is difficult process of finding the best decision. Active computerization of our intensive life forces mathematical apparatus of optimization to improve and generates new and new optimization problems. Optimization modeling is rather powerful tool, used, including, in classical mathematics. Today, with active use of computer technologies in our life, we have to admit that computer methods give us innovative way to look at different problems from classical mathematics. Even the simplest problems of linear algebra are not just routine tasks for students and

can be considered from a different angle via modern techniques. Linear algebra is, probably, the most fundamental tool for machine learning, providing indeed powerful and versatile framework for representing, analyzing, and manipulating data. Its broad applicability to truly wide spectrum of machine learning tasks makes it indeed indispensable skill for professionals in the corresponding field. Applied linear algebra allows us to take some different look at many classical problems of mathematics [2; 3], that is demonstrated in this paper.

It presents some results of comparison between traditional method [1] and optimization gradient methods, as well as their variations [4], for solving arbitrary (not only quadratic) systems of linear algebraic equations. These results have been obtained after testing our own program, written in «Visual Basic for Applications». Namely, we have combined well-known methods from classical algebra and the next two optimization methods: gradient descent method with adapted step selection and gradient descent method with adapted step correction.

Main part. As far as we know, solving an arbitrary system of the form

$$A\vec{x} = \vec{b},$$

where

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} = A_{m\times n},\ \vec{b} = \begin{pmatrix} b_1 \\ \dots \\ b_m \end{pmatrix} = B_{m\times 1},\ \vec{x} = \begin{pmatrix} x_1 \\ \dots \\ x_n \end{pmatrix} = X_{n\times 1},$$

reduces to finding minimizer, i.e. such vector $\vec{x}^* \in \mathbf{R}^n$, for which

$$\min_{\vec{x}\in \mathbf{R}^n} f(\vec{x}) = f(\vec{x}^*),$$

for the next residual function

$$f(\vec{x}) = \left(A\vec{x} - \vec{b}, A\vec{x} - \vec{b}\right) = \left\|A\vec{x} - \vec{b}\right\|^2,\ A \in \mathbf{R}^{m\times n},\ \vec{b} \in \mathbf{R}^m,\ m \le n,$$

where $\|\cdot\|$ denotes the Euclidean norm, $(\cdot,\ \cdot)$ – the Cartesian product. This minimum is reached at the solution $\vec{x}^* \in \mathbf{R}^n$ of the system under consideration.

Formally, we take one random starting point $\overrightarrow{x} \in \mathbf{R}^n$ and iteratively improve it. More precisely, we iteratively generate some sequence of such points

$\{\vec{x}_k\}|_{k\geq 0}$ (i.e. some descent trajectory, that is sometimes called a relaxation trajectory, that converges to our real solution $\vec{x}^*$), that

$$f(\vec{x}_{k+1}) \leq f(\vec{x}_k),\ k \geq 0,$$

according to some iterated scheme.

We have written the program that combines «classical» and «modern» methods for solving arbitrary systems of linear algebraic equations. It consists of two parts: the first is based on the Gaussian elimination method [1] and the second – on using various gradient methods [4]. Screenshots of the program's results, specifically, visual demonstrations of implementation of two gradient methods: gradient descent method with adapted step selection and gradient descent method with adapted step correction, – for the system

$$\begin{cases} x_1 + 3x_2 + 2x_3 = 10, \\ 4x_1 + 3x_2 + 2x_3 = 20, \end{cases}$$

are shown below. Namely, necessary «traditional» result of the Gaussian elimination method: the obtained triangular form of the system, – and «innovative» results of two variational methods: the obtained solution, accuracy of all calculations, and running time of the corresponding method.

The obtained triangular form of the system under consideration has been got with the help of traditional Gaussian elimination method. According to it, the proposed system can be rewritten in the next matrix form

$$\begin{cases} x_1 + 3x_2 + 2x_3 = 10, \\ 4x_1 + 3x_2 + 2x_3 = 20, \end{cases} \leftrightarrow \begin{pmatrix} 1 & 3 & 2 \\ 4 & 3 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 10 \\ 20 \end{pmatrix} \leftrightarrow A\vec{x} = \vec{b},$$

where $A = \begin{pmatrix} 1 & 3 & 2 \\ 4 & 3 & 2 \end{pmatrix}$ is the corresponding to the system coefficient matrix, $\bar{A} = \left(\begin{array}{ccc|c} 1 & 3 & 2 & 10 \\ 4 & 3 & 2 & 20 \end{array}\right)$ is the corresponding to the system augmented matrix, and $\vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$. With the help of elementary row operations for transforming this matrix into row-echelon form, we get the next result

$$\left(\begin{matrix}1 & 3 & 2\\ 4 & 3 & 2\end{matrix}\middle|\begin{matrix}10\\ 20\end{matrix}\right) = \left(\begin{matrix}1 & 3 & 2\\ 4 & 3 & 2\end{matrix}\middle|\begin{matrix}10\\ 20\end{matrix}\right)\begin{matrix}I\\ II\end{matrix} \to \left(\begin{matrix}1 & 3 & 2\\ 4 & 3 & 2\end{matrix}\middle|\begin{matrix}10\\ 20\end{matrix}\right)\begin{matrix}I\\ II-4\times I\end{matrix} =$$

$$= \left(\begin{matrix}1 & 3 & 2\\ 4-4 & 3-12 & 2-8\end{matrix}\middle|\begin{matrix}10\\ 20-40\end{matrix}\right)\begin{matrix}I\\ II\end{matrix} = \left(\begin{matrix}1 & 3 & 2\\ 0 & -9 & -6\end{matrix}\middle|\begin{matrix}10\\ -20\end{matrix}\right)\begin{matrix}I\\ II\end{matrix} \to$$

$$\to \left(\begin{matrix}1 & 3 & 2\\ 0 & -9 & -6\end{matrix}\middle|\begin{matrix}10\\ -20\end{matrix}\right)\begin{matrix}I\\ II\times(-1)\end{matrix} = \left(\begin{matrix}1 & 3 & 2\\ 0 & 9 & 6\end{matrix}\middle|\begin{matrix}10\\ 20\end{matrix}\right).$$

The last matrix represents the next equivalent system to the given one

$$\begin{cases} x_1+3x_2+2x_3=10,\\ 4x_1+3x_2+2x_3=20,\end{cases} \leftrightarrow \begin{cases} x_1+3x_2+2x_3=10,\\ 0+9x_2+6x_3=20,\end{cases} \to \begin{cases} x_1+3x_2+2x_3=10,\\ 9x_2+6x_3=20.\end{cases}$$

As far as we see,

$$rang\left(\begin{matrix}1 & 3 & 2\\ 0 & 9 & 6\end{matrix}\right) = rang\left(\begin{matrix}1 & 3 & 2\\ 0 & 9 & 6\end{matrix}\middle|\begin{matrix}10\\ 20\end{matrix}\right) = 2,$$

thus the system is solvable. Since the obtained rang (two) is less than the given quantity of the arguments (three), there exist infinitely many solutions. Back-substitution yields

$$\begin{cases} x_1+3x_2+2x_3=10,\\ 9x_2+6x_3=20,\end{cases} \to \begin{cases} x_1+3x_2+2x_3=10,\\ x_2=\dfrac{20-6x_3}{9},\end{cases} \to$$

$$\to \begin{cases} x_1+\dfrac{3(20-6x_3)}{9}+2x_3=10,\\ x_2=\dfrac{20-6x_3}{9},\end{cases} \to \begin{cases} x_1+\dfrac{20-6x_3}{3}+2x_3=10,\\ x_2=\dfrac{20-6x_3}{9},\end{cases} \to$$

$$\to \begin{cases} x_1+\dfrac{20}{3}-\dfrac{6x_3}{3}+2x_3=10,\\ x_2=\dfrac{20-6x_3}{9},\end{cases} \to \begin{cases} x_1+\dfrac{20}{3}-2x_3+2x_3=10,\\ x_2=\dfrac{20-6x_3}{9},\end{cases} \to$$

$$\to \begin{cases} x_1+\dfrac{20}{3}=10,\\ x_2=\dfrac{20-6x_3}{9},\end{cases} \to \begin{cases} x_1=\dfrac{10}{3},\\ x_2=\dfrac{20-6x_3}{9},\end{cases} \to \begin{cases} x_1=\dfrac{10}{3},\\ x_2=\dfrac{20-6x_3}{9},\\ x_3=x_3,\end{cases}$$

where $x_3 \in \mathbf{R}$ is free.

According to the proposed optimization methods, we have obtained the next results, demonstrated on the pictures below (fig. 1, fig. 2).

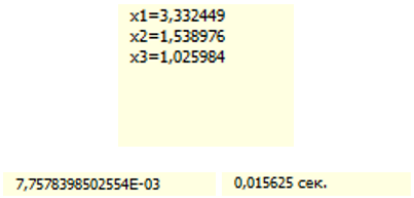


Fig. 1. The obtained results of gradient method with adapted step selection

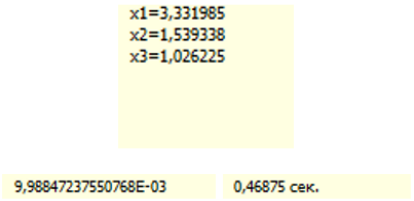


Fig. 2. The obtained results of gradient method with adapted step correction

Conclusions. Active computerization of our life forces mathematical apparatus of optimization to improve and generates new optimization problems. Optimization modeling is rather powerful tool and is ideally suited to classical linear algebra. We suppose the considered software product for solving linear systems via modern optimization methods to be an innovative support for studying the course of linear algebra and a cycle of various related disciplines.

References

1. Axler S. Linear Algebra Done Right (Undergraduate Texts in Mathematics). Springer, 2015. 340 p.

2. Meyer C. D. Matrix Analysis and Applied Linear Algebra. Society for Industrial and Applied Mathematics (SIAM), 2010. 730 p.

3. Shores T. S. Applied Linear Algebra and Matrix Analysis (Undergraduate Texts in Mathematics). Springer, 2018. 479 p.

4. Бейко І. В., Зінько П. М., Наконечний О. Г. Задачі, методи і алгоритми оптимізації. Рівне: НУВГП, 2011. 624 с.