The Ministry of Education and Science of the Russian Federation

Lobachevsky State University of Nizhni Novgorod

Computing Mathematics and Cybernetics faculty

The competitiveness enhancement program

of the Lobachevsky State University of Nizhni Novgorod

among the world's research and education centers

Strategic initiative

“Achieving leading positions in the field of supercomputer technology

and high-performance computing”

ITERATIVE METHODS FOR SOLVING LINEAR SYSTEMS

Lecture 1. Introduction. Basic iterative methods.

Nizhni Novgorod

2014

OBJECTIVES

The objective of this lecture is to review general concepts of solving linear systems using basic iterative methods (Jacobi, Seidel and Successive Over Relaxation (SOR) methods).

ABSTRACT

This lecture will deal with basic iterative methods of solving linear systems (i.e. Jacobi, Seidel and SOR methods) and introduce the respective theoretic concepts such as convergence and convergence rate. We shall discuss parallel versions of the mentioned algorithms as applicable to dense and sparse systems as well as one of the means to accelerate iterative methods. Chebyshev polynomials will be used as principal mathematical tools. A computational procedure based on the best properties of these polynomials will be used to accelerate iteration convergence.

GUIDELINES

This time we will start considering methods of solving linear systems that are fundamentally different from direct ones, i.e. exact problem solving methods. Let $Ax=b$ be a system of linear equations related to an unknown vector $x \in R^n$ with a $n \times n$ symmetric positive definite (SPD) matrix size A and vector $b \in R^n$ as the right-hand side vector. Let x^* be the exact system solution.

An iterative method of solving linear systems generates a sequence of vectors $x^{(s)} \in R^n$, $s=0, 1, 2,\ldots$; the researcher may consider each of them as an approximate system solution. The first approximation, vector $x^{(0)} \in R^n$, is determined by the researcher. The method is classified as a *convergent* one, if for any first approximation $x^{(0)} \in R^n$ the sequence $x^{(s)} \in R^n$, $s=0, 1, 2,\ldots$, converges to the exact solution x^*. As a matter of practice, two criteria for stopping iterations are used: accuracy and number of iterations.

Iterative methods are used to solve both dense and sparse linear systems. Unlike direct methods, iterative ones do not lead to matrix filling in case of sparse systems. As for the implementation, it involves only multiplying a matrix by a vector thus making sparse systems a preferable application of iterative methods.

The first part of this lecture will deal with the fixed point iteration method determined by $x^{(s+1)} = Gx^{(s)} + c$, where G is a iteration matrix. This lecture will also formulate the method and give a list of its main properties. In particular, a necessary and sufficient condition of convergence is the inequation $\rho(G) < 1$, where $\rho(G)$ is the *spectral radius* of the matrix G. We will also formulate the parametrized fixed point iteration method described as $(x^{(s+1)} - x^{(s)})/\tau + Ax^{(s)} = b$. Method properties will be formulated depending on the parameter

(convergence domain and best parameter choice). Method parallelization issues will also be discussed.

The second part of this lecture will deal with classic iterative methods based on the idea to represent the matrix A as a sum of three matrices, $A=L+D+R$. Here, D is a $n \times n$ diagonal matrix whose principal diagonal coincides with that of the matrix A. L is the lower $n \times n$ triangular matrix whose non-zero (below-diagonal) elements also coincide with those of A and its principal diagonal is the zero one. Similarly, R is the upper $n \times n$ triangular matrix whose non-zero (above-diagonal) elements coincide with those of A and its principal diagonal is the zero one, too.

This lecture will also formulate Jacobi, Seidel and SOR methods. In a matrix format, these methods look like $Dx^{(s+1)} = (-L - R)x^{(s)} + b$ (Jacobi method), $(L + D)x^{(s+1)} = -Rx^{(s)} + b$ (Seidel method), $\frac{(D + \omega L)(x^{(s+1)} - x^{(s)})}{\omega} + Ax^{(s)} = b$ (SOR). The methods will be represented component-wise; the problem of the best ω value selection for the SOR method will also be discussed. It will be shown that the SOR method is the most efficient among the described ones (when the respective ω value is selected). We will discuss the ways to parallelize the SOR method and see the results of the use of SOR for solving linear systems with a dense matrix.

The next part of this lecture will deal with issues related to search acceleration. Let us solve the $Ax=b$ system using any iterative method (e.g. SOR) with the iteration matrix G. If the process is convergent, i.e. $\rho(G) < 1$, the resulting sequence of vectors will converge to the true solution x^*. Now let us suppose that m iterations of the selected method took place, resulting in vectors $x^{(0)}, x^{(1)}, ..., x^{(m)}$, each being an approximation of x^*. The problem is to find the linear combination $y^{(m)}$ of these vectors that will approximate x^* better than $x^{(m)}$. This problem is reduced to minimization of the matrix spectral radius $p_m(G)$, where p_m is a polynomial.

Let us suppose that the transition matrix G has the following properties (which is routinely satisfied for a number of iterative methods):

- All matrix G eigenvalues are real;
- All these eigenvalues are within $[-\rho,\rho]$, where $0<\rho<1$.

Therefore, we can try to obtain the polynomial p_m, where $p_m(1) = 1$; and $\max_{-\rho<x<\rho} |p_m(x)|$ has the smallest possible value among all m^{th} polynomials.

As the matrix $p_m(G)$ eigenvalues are $p_m(\lambda(G))$, such eigenvalues will be small, and so will be the spectral radius (as the largest among the eigenvalues moduli). Generation of the polynomial p_m that meets the formulated conditions is a classic problem of the approximation theory

whose solution is based on *Chebyshev's polynomials*. Using Chebyshev's polynomials, we may propose an effective method of coefficient computation for the linear combination $y^{(m)}$ without storing all vectors $x^{(0)}, x^{(1)}, ..., x^{(m)}$.

Unfortunately, the algorithm above is not directly applicable to the SOR method used for solving the $Ax=b$ system. The point is that the iterative process matrix G for the SOR method has, in general, complex eigenvalues, while Chebyshev's acceleration requires that the G eigenvalues be real and belong to $[-\rho,\rho]$. However, this situation may be remedied using the Symmetric SOR method (SSOR). As the name implies, this method is a SOR method modification. One SSOR stage to calculate the $(s+1)^{th}$ approximation of $x^{(s+1)}$ consists of two steps:

- A SOR step that involves calculation of the intermediate approximation $x^{(s+1/2)}$ components in the normal order;
- A SOR step that involves calculation of the new approximation $x^{(s+1)}$ components in the reverse order.

For SSOR with Chebyshev's acceleration, computational experiments were performed to compare the resulting convergence rate with that of the initial method. As expected, this method proved to be the best if used for solution of a test problem with known eigenvalues of the matrix A and transition matrix G. Plus, a tenfold acceleration is observed if compared to SOR.

RECOMMENDATIONS FOR STUDENTS

A description of basic iterative methods may be found in any numerical method manual, e. g. [1, 2]. [3] details the Chebyshev acceleration method.

REFERENCES

1. David R. Kincaid and E. Ward Cheney. Numerical analysis : mathematics of scientific computing. Brooks/Cole Publishing Company, 1991.
2. Richard L. Burden, J. Douglas Faires. Numerical Analysis. Brooks Cole, 2000.
3. James W. Demmel. Applied Numerical Linear Algebra. SIAM, 1997.

PRACTICE

1. Implement the fixed point iteration method to solve a linear system with a SPD matrix (both its sequential and parallel versions). Estimate how the method parameter influences the convergence rate. Study the parallel algorithm scalability.

2. Implement the Jacobi and Seidel methods to solve a system of linear equations with a SPD matrix. Compare the respective convergence rates. Propose parallel implementations of the above algorithms.
3. Implement the SOR method to solve a system of linear equations with a SPD matrix. See how the method parameter influences the convergence rate. Propose a parallel implementation and study its scalability.

TEST

1. For linear systems with a SPD matrix, the SOR method will converge at
    a. $+\omega\in(0, 2)$
    b. $\omega\in(-2, 2)$
    c. any ω value
2. For linear systems with a SPD matrix, the Seidel method will converge at
    a. a zero first approximation only
    b. a unit norm first approximation only
    c. + any first approximation
3. For linear systems with a SPD matrix, the fixed point iteration method will converge at
    a. $\tau\in(0, 2/\lambda_{max})$
    b. + $\tau\in(0, 2/(\lambda_{min}+\lambda_{max}))$
    c. any τ value
4. For the fixed point iteration method,
    a. a parallel algorithm cannot be created
    b. the solving process may be parallelized at all
    c. + a single iteration may be parallelized
5. Jacobi method may be effectively parallelized
    a. for small dense matrices
    b. for large dense matrices
    c. + for any dense matrices
6. SOR method may be effectively parallelized
    a. for small dense matrices
    b. +for large dense matrices
    c. for any dense matrix
7. SOR method
    a. may be effectively parallelized for sparse matrices of any structure
    b. + may be effectively parallelized for large sparse block diagonal matrices

c. cannot be parallelized for sparse matrices

8. Iteration matrix eigenvalues for the SSOR method are
    a. complex in general
    b. real within the range $[-\rho,\rho]$, where $\rho>1$
    c. + real within the range $[-\rho,\rho]$, where $0<\rho<1$
9. Chebyshev's acceleration may be applied to
    a. Any iterative method
    b. +An iterative method with a symmetric iteration matrix
    c. Any iterative method used to solve linear systems with a symmetric matrix
10. Chebyshev's acceleration may be applied if
    a. +the spectral radius of the method iteration matrix is estimated
    b. the spectral radius of the system matrix is estimated
    c. no spectral radius estimation is required.