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

Practice 5. Solving sparse linear systems
using the preconditioned generalized minimum residual method

Nizhni Novgorod

2014

OBJECTIVES

The purpose of this laboratory work is to demonstrate practical implementation of the generalized minimum residual method and study influence of $ILU(0)$-preconditioning on the method convergence rate.

ABSTRACT

This laboratory work reviews one of the iterative methods of solving general sparse linear systems, i.e. the generalized minimum residual (GMRes) method. In the course of work, it is proposed to implement this method for linear systems with non-symmetric sparse matrices. The condition number μ_A of the linear system matrix will have a decisive influence on the method convergence rate, so the next task will consist in studying the method convergence depending on μ_A. Experiments prove poor convergence for ill-conditioned problems. Then it is proposed to use the $ILU(0)$-preconditioner to improve the linear system matrix condition and analyze the preconditioned method convergence.

GUIDELINES

The laboratory work is intended to implement the generalized minimum residual (GMRes) method for solution of the linear system $Ax = b$, where A is a general sparse matrix. The system matrix is stored in the CRS format.

In the beginning, we shall substantiate computation formulas used for solving linear systems by the GMRes method and describe the general iterative scheme of the solution. The method is based on approximation of the exact system solution by the vector x_m of the Krylov subspace $x_0 + K_m$ which minimizes the residual of the linear system $r = \|b - Ax\|$. The approximated vector is computed as $x_m = x_0 + V_m y_m$, where V_m is a matrix composed by the Krylov subspace orthonormal basis vectors $K_m = span\{b, Ab, A^2b, \dots, A^{m-1}b\}$. The vector y_m can be found as a solution of an overdetermined linear system in the least square sense $y_m = \arg\min_y \|\beta e_1 - \bar{H}_m y\|$. For this purpose, the matrix of the auxiliary system $\bar{H}_m y = \beta e_1$ is reduced to the upper triangle by means of Givens rotations. Then we shall consider a modified iterative scheme that involves the use of the ILU(0)-preconditioner to reduce the method computational error. In this case, the formulas to compute the initial residual vector and coefficients of the auxiliary overdetermined linear system are subject to modification.

Sequential implementation is strictly based on the computation formulas. It will be proposed to compare this implementation to the performance of the Intel MKL iterative solver involving the Flexible Generalized Minimum Residual (FGMRes) method which is a modification of the GMRes method. Default initialization of the MKL solver and its parameters is performed by

dfgmres_init(); the set parameters are checked for correctness by **dfgmres_check()**; the FGMRes method iterations are performed by means of **dfgmres()**. Method results can be obtained with the help of **dfgmres_get()**. In the course of computation, renewal of auxiliary data will require calling the **mkl_dcsrgemv()** function to multiply a sparse matrix by a vector.

Then students will be proposed to modify the consecutive implementation to improve the method convergence by means of the *ILU*(0)-preconditioner whose construction is implemented in the laboratory work on the Preconditioner Construction Using Incomplete *LU*-factorization. This will require implementation of the backward Gaussian method to solve sparse linear systems with a triangular matrix. In a similar way, the linear system solution will be modified using the MKL library.

The laboratory work will list experimental results for matrices from the University of Florida Sparse Matrix Collection [3]. Condition numbers of two of the five matrices were less than 5,000 while those of the remaining three matrices exceeded 10^9. For the purpose of experiments, the absolute residual norm precision has been selected as the method stop criterion.

As experimental results show, without preconditioner both the implemented and the MKL library method would not converge for any of the considered matrices. In other cases, these methods converge at a close number of iterations (up to 35 iterations for well-conditioned matrices and up to 78 iterations for ill-conditioned ones). If the required method accuracy is reduced by 4 orders of magnitude for well-conditioned matrices, the number of method iterations will grow by maximum 10. For matrices with a high condition number, the convergence rate reduces as the accuracy increases. For most matrices, the use of preconditioner helps reduce the iterations count 3 through 5 times. For the last matrix, the MKL library FGMRes method will converge in 20 through 22 iterations if the ILU-preconditioner is used; if implemented on its own, this method converges only at a greater required accuracy.

RECOMMENDATIONS FOR STUDENTS

A brief method description including algorithm pseudocode can be found in [1]. A detailed description of the method including the respective conclusions, theoretical justification and application examples can be found in [2].

REFERENCE

1. J. Dongarra et al. Templates for the solution of linear systems: building blocks for iterative methods. SIAM, 1994.
2. Y. Saad. Iterative Methods for Sparse Linear Systems. SIAM, 2003.
3. The University of Florida Sparse Matrix Collection – [www.cise.ufl.edu/research/sparse/matrices/]

PRACTICE

1. Use the $ILU(p)$-preconditioner for the GMRes method. Analyze the convergence, compare the number of iterations and method convergence for the $ILU(0)$ and $ILU(p)$ preconditioners.
2. Implement the GMRes(m) method (generalized minimum residual method with restart). Compare the number of method iterations for various restart parameter m values.

TEST

1. What systems are solvable using the generalized minimum residual method?
   a. Only systems with a symmetric positive definite matrix
   b. Only systems with a symmetric matrix
   c. + Systems with a general matrix
2. How to define the Krylov subspace K_m ?
   a. + $K_m = span\{b, Ab, A^2b, \dots, A^{m-1}b\}$
   b. $K_m = span\{b, Ab, A^2b, \dots, A^m b\}$
   c. $K_m = span\{Ab, A^2b, \dots, A^m b\}$
3. How many generalized minimum residual method iterations is required to guarantee convergence for solving the system $Ax=b$ with the matrix A sized $n \times n$?
   a. + n
   b. $O(n)$
   c. n^2
4. Which computing operation of the GMRes method is most complex?
   a. Finding matrix eigenvalues
   b. Multiplication of two sparse matrices
   c. +Solving a linear system by the least squares method
5. In the GMRes method, the system solution is approximated by the Krylov subspace vector $x_m = x_0 + V_m y$, where $V_m \dots$
   a. Matrix from the system A submatrix eigenvectors
   b. +Krylov subspace orthonormal vector basis

c. Krylov subspace orthogonal vector basis

6. How are nonzero coefficients of the Givens rotation matrix Ω_i selected?
   a. +to zero the coefficient $h_{i+1,i}$
   b. to zero the coefficient $h_{i,i}$
   c. to zero the coefficient $h_{i,i+1}$
7. What computation formulas of the method remain unchanged when the preconditioner is used?
   a. Initial residual computation
   b. +Computation of coefficients for the Givens rotation matrix Ω_i
   c. Computation of coefficients for the auxiliary system $\overline{H}_m$
8. How many calls for the **`dfgmres()`** function will be required to solve a sparse system with a matrix sized $n \ x \ n$?
   a. One
   b. n
   c. +Several but not more than .
9. If the output parameter of **`dfgmres()RCI_request`** is equal to 2, it means that:
   a. +Stop criterion check is required
   b. Current orthogonal vector norm has to be checked for zero
   c. The function has completed operation with an error
10. What does **`dfgmres_check()`** do?
    a. Checks the solution of the system $Ax = b$ for correctness
    b. +Checks the set parameters of the iterative solver for correctness
    c. Checks GMRes iterations for correctness