IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE)
e-ISSN: 2278-1684,p-ISSN: 2320-334X, Volume 17, Issue 5 Ser. II (Sep. – Oct. 2020), PP 48-57
www.iosrjournals.org

An Overview of Iterative Techniques Used for Solving the Systems of Equations Resulting from the Finite Element Analysis

Baher A. Haleem[1], Ihab M. El Aghoury[2], Bahaa S. Tork[3], Hisham A. El-Arabaty[4]

[1](Department of Structural Engineering, Faculty of Engineering/ Ain Shams University, Cairo, Egypt)
[2](Department of Structural Engineering, Faculty of Engineering/ Ain Shams University, Cairo, Egypt)
[3](Department of Structural Engineering, Faculty of Engineering/ Ain Shams University, Cairo, Egypt)
[4](Department of Structural Engineering, Faculty of Engineering/ Ain Shams University, Cairo, Egypt)
Corresponding Author: Baher A. Haleem

Abstract:*The sophistication and size of the models of the finite element method are continually growing. Hence, there is a rising need for faster solvers. The speed of any algorithm varies depending on the analyzed system's scale and the features of the coefficient matrix, that accordingly affect the selection of the proper solver depending on various standards, for instance, the required storage, the results' accuracy, and solving speed. However, these demands are, unfortunately, usually contradictory; there is no single procedure which outperforms the other techniques in all cases. The two main classes of solving techniques are Direct and Iterative (Indirect) solvers. Generally, direct methods tend to need many computations and large memory space, especially for significantly large problems, so a long time is elapsed during the analysis process. Consequently, an iterative solver, in such cases, is more desirable. Besides, such solvers are generally simpler to program. The main objective of this study is to provide a historical background in addition to shedding light on the latest research literature of several well-known classical and modern iterative techniques that have been used in solving myriad engineering problems. In this paper, the difference between both classes of the direct and iterative finite element solvers is explained; showing their strengths and weaknesses, besides mentioning the proper cases to use each class. Some examples of well-known direct and indirect techniques are mentioned. Then, attention is paid to the iterative techniques till the end of the paper; the eminent methods within the two main classes of iterative solvers, classical and modern methods, are mentioned with providing a historical background and literature review for them.*

Keywords: *Finite element method, Direct solvers, Indirect solvers, Classical iterative methods, Modern iterative methods*

Date of Submission: 25-09-2020 Date of Acceptance: 08-10-2020

I. Introduction

Among the famous numerical techniques for solving various engineering problems, is the finite element method (FEM), where we solve a set of matrix linear equations in the form of [K]{u} = {f}. Here [K] refers to the stiffness matrix of the structure, which is the coefficient matrix, {f} refers to the forces vector applied to the structure, and {u} refers to the deformations vector which is the set of unknowns to solve for[1]. In general, stiffness matrices are symmetric matrices of order n×n which depends on the number of DOFs in the structure that may be thousands to millions of DOFs. Reaching the solution to the above-mentioned system of linear equations is usually the most time-consuming and exhausting part through the entire process[2].

The use of iterative equation solvers in commercial finite element software has been steadily growing. Performing a finite element analysis is typically accomplished in three stages, preprocessing, analysis, and postprocessing. The analysis stage is by far the most time-consuming, and the efficiency of the equation solver utilized is a decisive factor in determining the computation time and storage requirements for the analysis, and therefore has a significant effect on the analysis costs, especially in case of large problems.

II. Solvers Classification

There are two main classes of solvers (algorithms) that are used to find the values of the unknowns which are Direct methods and Iterative (Indirect) methods[2], [3].

2.1 Direct methods

The direct methods find the exact values of the unknowns where there are no errors except the error of the round-off due to the machine [4]. The direct solution methods are guaranteed to reach a solution, as long as the model of the structure is set up properly[5].

A direct solver is commonly used in structural analysis using commercial packages because finding solutions using direct solvers is stable without being influenced by the coefficient matrix numerical characteristics so the direct solver can be more robust than the iterative solver[2]. However, it generally tends to need remarkably large storage demands and a great amount of calculation especially for huge problems because a large number of equations are solved simultaneously. Thus, it spends a long run time. Accordingly, in these cases, an iterative solver that requires relatively less memory space is more desirable[3].

Depending on the way of the connection of the elements together, the coefficient matrix, i.e., stiffness matrix [K] is usually very sparse;a lot of the items have a value of zero[5]. What is usually cared about when using a direct solver, is the proper usage of the sparsity of the stiffness matrix, [K]. Since a zero value has no contribution to the solution, so most solution methods are programmed in some approach or another to disregard the zero terms. The amount of computation and storage requirements significantly vary depending on the technique of utilizing the sparsity[3]. A method may need a small amount of memory space to store the matrix, however, require optimization to attain the least demand for memory space, whereas another method may use more variables for matrix storage, and hence need more memory space, but be faster overall[5].

There are different techniques for direct methods such as Gaussian elimination, Gauss-Jordan Elimination, and other decomposition methods like Cholesky decomposition, Incomplete Cholesky factorization, LU factorization, LDU decomposition, LDL^T decomposition, QR decomposition, etc.

Sometimes, using the elimination technique is annoying especially with large sparse problems [6] because the matrix sparse characteristics are lost since it results in some elements, originally zero, becoming nonzero as elimination proceeds. Not only does this increase the storage problems on computers, but also computing times for elimination are increased[7].

2.2 Iterative methods

Iterative solvers are used mostly in research related programs, especially in nonlinear analysis. Most commercial packages utilize direct solvers, while some commercial packages include additional iterative solvers.

In the iterative solvers, an initial guess for the solution is assumed. By substituting this assumed solution in the system of linear equations, a better estimate is attained for each unknown. Then, this process is successively repeated until the solution reaches an almost constant value, i.e., no longer changes[5]; hence, the errors are minimized until convergence is reached through iterative calculations. It is crucial to quickly reduce the convergence errors through a small number of iterations[3]. For well-conditioned problems, the convergence should be reasonably monotonic. If the problems are not well-conditioned, thus the convergence shall be slower. Oscillatory performance of an iterative solver often indicates that the problem isn't appropriately set up, for example, when the problem isnot sufficiently constrained[2].

Iterative methods, in contrast to direct methods, reach the solution gradually step by step, instead of through one large computational step[2]. Eventually, iterative methods result in approximate solutions where some residual errors exist; the values of these residual errors depend on the demanded accuracy. The solution accuracy is regulated by the tolerance value of the convergence; a smaller tolerance results in a solution with higher accuracy but it may take more iterations[5].

The iterative solvers have a great advantage which is their memory usage, as it is significantly less than that of direct solvers for problems of the same size[2]. Iterative solvers, generally, are simpler to program. They are also capable of solving an n x n system of linear equations without finding the matrix inverse of the coefficient matrix, stiffness matrix [K] in finite element, and without needing to store the matrix [K] entirely, which results in saving the time requirements and memory space. However, it should be cautioned, as mentioned in [3], that iterative solvers may not result in the desirable solutions on account of the coefficient matrix numerical characteristics, or the quantity of iterative calculation may possibly become significant in reaching converged solutions[3]. Thus, the iterative solvers are considered the most appropriate procedures for solving huge models of sparse and large coefficient matrices, providing they are able to converge[5].

It can also be possible to carry out quick re-solutions in which small structural modifications in the system have been made, through starting the iterations with the values of the variables obtained from the analysis conducted for the original structure[7].

When using an iterative solver, sometimes a preconditioning procedure is applied before starting the iterative solution process to improve the stiffness matrix condition number. This can make a change to the entire behavior of the system depending on the preconditioning technique[1], [3]. A system having a low condition number means well-conditioned, while a high condition number means ill-conditioned[8].

The iterative techniques are commonly divided into two main categories which are:

- Classical (Stationary) (Relaxation) methods: Jacobi, Gauss-Seidel method, Successive Overrelaxation (SOR), Symmetric Successive Overrelaxation (SSOR), Moment Distribution by Hardy Cross, and Rotation Contribution method (Kani's Method).
- Modern methods (Krylov subspace methods): steepest descent, Conjugate Gradient method (CG), Generalized minimal residual (GMRES), Minimal residual method (MINRES), Biconjugate Gradient (BiCG), and Multigrid method.

III. Iterative Solvers Overview

The performance of a solver changes depending on the size or scale of the system that is meant to be analyzed[3] as well as the properties and characteristics of the coefficient matrix, that in turn affect the selection between the solvers. There is no single procedure which outperforms the other techniques in all cases.

A method is evaluated whether it is worthy or not depending on some standards such as the accuracy of the solution worked out, the required amount of computations, and the required storage on the computer. Unfortunately, these necessities are usually contradictory, thus it is still critical to realize new effectual solutions.

Although the single iteration of the classical iterative methods generally takes a shorter time than that of the modern methods, as classical methods have simpler algorithms (steps), the modern methods are generally faster as they take a fewer number of iterations till convergence.

3.1 Classical iterative methods

Numerous researches had been conducted to study and scrutinize many classical iterative techniques. Some of these techniques are discussed hereunder.

3.1.1 Jacobi and Gauss-Seidel (GS) methods

The Jacobi iterative method is named after the German mathematician Carl Gustav Jacob Jacobi (1804–1851). The Gauss-Seidel technique, which is the modification of Jacobi technique, is named after both Carl Friedrich Gauss (1777–1855) and Philipp L. Seidel (1821–1896).

A comparison between Jacobi iterative technique and Gauss-Seidel iterative technique was conducted by A. I. Bakari and I. A. Dahiru [9]. The Jacobi iterative method makes two assumptions; the first one is that the given system has a unique solution, while the second one is that the coefficient matrix has no zero items on its main diagonal. They mentioned that if any of the entries of the diagonal is zero, then the rows or columns should be interchanged to get a coefficient matrix which has no zero entries on its main diagonal.

In the Jacobi technique, the obtained values in the n^{th} iteration remain unchanged till the entire n^{th} iteration has been calculated. On the other hand, in the Gauss-Seidel technique, the new values of the variables obtained in the n^{th} iteration are used in the same iteration once they are known. That is, as soon as we have obtained from the first equation the value of the first variable, this value is then utilized in the second equation in order to get the new value of the second variable. Similarly, the new value of the second variable and that of the first variable are utilized in the third equation in order to get the new value of the third variable and so on[9].

Moreover, A. I. Bakari and I. A. Dahiru [9] discussed the convergence of these two methods. The convergence rate of iterative techniques shows how fast the error approaches zero with the increase in the number of iterations.

The coefficient matrix [A] is a convergence matrix provided that $\rho(A) < 1$, where $\rho(A)$ is the spectral radius of [A]. This condition is achieved for both Jacobi method and Gauss-Seidel method on condition that the coefficient matrix [A] is diagonally dominant. The matrix is called a diagonally dominant matrix in case that the absolute value (magnitude) of the entry on the diagonal in each row is larger than or equal to the sum of absolute values (magnitudes) of non-diagonal entries in the same row, as provided by (1). Moreover, the matrix is strictly diagonally dominant provided that the condition shown in (2) is valid [10]:

$$\sum_{j \neq i} \left| a_{ij} \right| \leq \left| a_{ii} \right| \qquad (1)$$

$$\sum_{j \neq i} \left| a_{ij} \right| < \left| a_{ii} \right| \qquad (2)$$

Where; a refers to terms of the coefficient matrix [A], i is the row number in [A], and j is the column number in [A].

Their results showed that Gauss-Seidel iterative technique outperforms Jacobi iterative technique with respect to the accuracy and the required number of iterations for convergence to occur; Gauss-Seidel is more accurate and converges faster than Jacobi.

Liu Hongxia and Feng Tianxiang [11] studied the convergence conditions of Jacobi method and Gauss-Seidel method. After that, the iterative times of both methods were discussed. Then, the formula for the

estimated iterative times was obtained, because beforehand there were obvious differences between the estimated iterative times and actual iterative times [12]. Lastly, a numerical example was calculated by both methods and the results showed the estimated iterative times and the actual iterative times were basically equal.

Davod Khojasteh Salkuyeh [13] mentioned that both Jacobi and Gauss-Seidel iterative algorithms are considered stationary iterative techniques for solving a system of linear equations. There are some modern, popular iterative techniques such as GMRES[14] and Bi-CGSTAB[15] algorithms which are more effectual than Jacobi and Gauss-Seidel iterative methods. However, when these stationary methods are combined with the methods which are more efficient, e.g., as a preconditioner, they can be quite successful. Davod proposed a generalization of both Jacobi and Gauss-Seidel methods and studied their convergence. Then, he gave some numerical experiments to show their efficiency. It has been shown that if the coefficient matrix [A]is irreducibly diagonally dominant or if it is strictly diagonally dominant (SDD), the associated iterations of Jacobi method and Gauss-Seidel method converge for any initial guess [16]. Also, if [A]matrix is symmetric positive definite (SPD), the Gauss-Seidel approach also converges for any initial guess [17].

Finally, Davod concluded that the new techniques "Generalized Jacobi (GJ) method and Generalized Gauss-Seidel (GGS) method" are convenient for sparse matrices like matrices that arise from discretization of partial differential equations (PDEs). Also, his numerical results showed that the new generalized techniques have been more effectual than conventional Jacobi method and Gauss-Seidel method.

3.1.2 Successive Overrelaxation (SOR) method

Successive Overrelaxation (SOR) was developed by both H.Frankel [18] and David Young [19] in 1950. It was developed by adding a modification to the iteration model of Gauss-Seidel. SOR iterative algorithm is considered as one of the effective stationary iterative methods for solving a system of equations which is sparse and of large scale.

Ruixia Cui and Mingjun Wei [20] discussed astringency judgment conditions for Successive Overrelaxation (SOR) iterative algorithm and the importance of the proper selection of the convergence factor (ω). They provided then a MATLAB program based on the SOR iterative algorithm. They proved that the convergence rate of SOR iterative technique is faster than that of the Jacobi and Gauss-Seidel iterative techniques. They mentioned that the SOR iterative technique can be broadly applied in practice, essentially used for finding solutions to sparse systems of linear equations, so greatly decreasing computations and required internal memory of the computer. Accordingly, calculation efficiency increases.

T. Mayooran and Elliott Light [21] took a look at the basics of Successive Overrelaxation iterative method. Then, they applied the SOR method to a real-world problem which is solving the heat-equation while constant boundary temperature is being applied to a flat plate.They mentioned that SOR iterative method has applications in linear elasticity, mathematical programming, Fluid Dynamics and machine learning, etc. Among the cases of applications of SOR iterative technique in Dynamics are the study of turbulent flows, steady heat conduction, chemically reacting flows or boundary layer flows.

For [A] {x} = {b}, that is a system of linear equations, where {b} is the constants' vector, {x} is the unknowns' vector, [A] is the coefficient matrix;

- The iterative formula of Jacobi iterative method is given by (3):

$$x_i^{k+1} = \frac{1}{a_{ii}}\left(b_i - \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} x_j^{k}\right), \qquad (3)$$

$$i = 1, 2, ..., n$$

- The iterative formula of Gauss-Seidel iterative method is given by (4):

$$x_i^{k+1} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij} x_j^{k}\right), \qquad (4)$$

$$i = 1, 2, ..., n$$

- The iterative formula of Successive Overrelaxation iterative algorithm is given by (5):

$$x_i^{k+1} = (1-\omega) x_i^{k} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij} x_j^{k}\right), \qquad (5)$$

$$i = 1, 2, ..., n$$

Where; n is the number of unknowns; k is the order of the iterations; i and j are the order of rows and order of columns in matrix [A] respectively; x, a, and b refer to terms of the unknowns vector, the coefficient matrix, and the constants vector respectively. ω is called Relaxation Factor. The value of ω is generally greater than unity,

however, the successive overrelaxation (SOR) formula (5) gives the formula (4) of the Gauss-Seidel technique when ω equals unity.

The convergence is improved because the value of the variable at each particular iteration is formed of a combination of the newly calculated value and the old value. In other words, the SOR finds the new estimated value through multiplying the difference between both the new value and the preceding one by the relaxation factor (ω), then adding this scaled difference to the preceding value. Hence, the SOR iterative method resembles the Gauss-Seidel iterative method except that SOR uses the scaling factor (ω) to lessen the approximation error.

When ω equals unity, as mentioned previously, the SOR formula turns into the formula of Gauss-Seidel method. However, when ω is less than unity, it leads to the under-relaxation method, while at ω greater than unity, it becomes the over-relaxation method.

Generally, the value of relaxation factor ω required to attain a minimum number of iterations ranges from 1 to 2; but this value cannot be selected in advance, apart from some special cases [21]. The under-relaxation method (at $\omega < 1$) usually requires a larger number of iterations compared to that of the Gauss-Seidel method. Nevertheless, the under-relaxation method is sometimes needed to reduce the rate of convergence in the cases when a value of scaling factor ω that is greater than or equal to unity leads to divergence.

If [A] is a symmetric positive definite matrix, then the convergence is guaranteed with the successive overrelaxation technique for any chosen initial value of $x_0$as long as $0 < \omega < 2$ [21].

3.1.3 Accelerated overrelaxation (AOR) method

Apostolos Hadjidimos [22] clarified the Accelerated Overrelaxation (AOR) method which is a technique for the numerical solution of the systems of linear equations. The technique is a two-parameter generalization of the SOR technique such that when the two involved parameters are equal, the developed technique coincides with the Successive Overrelaxation (SOR) method. Finally, he gave a numerical example to show the superiority of AOR method.

Hadjidimos mentioned that the powerfulness of AOR compared to the other well-known techniques, such as SOR, is due to the presence of two parameters rather than usually at most one. Full exploitation of the existence of the two parameters would provide approaches that would converge faster than other techniques of the same type. Nevertheless, he mentioned that the determination of the optimum values of the acceleration and overrelaxation parameters was still a matter that needs further investigation.

Furthermore, sufficient conditions for convergence of the Accelerated Overrelaxation (AOR) technique have been considered by many other authors. In addition, to improve the rate of convergence of the AOR technique, the preconditioned AOR (PAOR) technique has been considered by several authors.

3.1.4 Kani's Method and Moment Distribution method

3.1.4.1 Historical background and relaxation concept

Classical structural analysis methods such as the Rotation Contribution method (Kani's Method) and Moment Distribution Method (Hardy Cross method) can be handy in quick and approximate analyses. They can be used for structures' primary analysis and also controlling the results of the computer programs.

3.1.4.1.1Analysis by Moment Distribution method

In 1930, Hardy Cross [23] proposed the Moment Distribution method that is also known as method of Hardy Cross. From the 1930s and until the computers began to be extensively used in the analysis and design of structures, the Hardy Cross method was the most broadly practiced method.

It provides a convenient tool for analyzing statically indeterminate structures, e.g., Beams and Frames, through manual calculations. The Cross approach has been taught in many universities as it has an easy interpretation. The method takes only flexural effects into consideration and disregards axial and shear effects. It could be used for simple programming in structural analysis, where the end moments of members are considered as the unknowns. This is principally an iterative process. It does not involve solving a system of simultaneous equations, as in the slope deflection method, providing the structures do not undergo transverse displacements, i.e., Sway motion.

It includes artificially restraining temporarily all rigid joints against rotations and calculating the bending moments produced by external loads, known as fixed end moments, and that is for all members. These moments at the joints of the structure in the original case without restraints are unbalanced. So as to equilibrate the joints, they are then released successively one by one. The unbalanced moments are then distributed at each released joint to all ends of the members interconnected at that joint proportionally to the corresponding stiffness of members. Certain factors of the distributed moments are transferred to the other end of each member (temporarily, rotationallyrestrained end) forming what is called carried-over moment.

Then, the released joint becomes restrained temporarily again before proceeding to the following joint. The same operations are applied at each joint until all the joints are concluded;thus, one cycle of operations is completed. The cycle is then repeated successively for a number of times until the values of the unbalanced

moments become negligible or till the obtained values are within the required accuracy. The final end moments of the members are the summation of all the distributed incremental moments [24]. End shear forces for the members are also obtained through applying the static equilibrium equations.

The method of Moment Distribution is also a displacement method for structural analysis. The Moment Distribution method in case of sway structures requires forming algebraic equations and solving them with a fewer number of unknowns.

3.1.4.1.2 Analysis by Kani's Method compared to Moment Distribution

Gasper Kani [25] introduced the Rotation Contribution method that is also known as Kani's Method. This method uses an iterative technique to solve the system of equations developed by the slope deflection method [26].

The Rotation Contribution method is self-correcting, to be exact, the error in a cycle—if any—is automatically corrected in the subsequent cycles provided that the distribution factors and fixed end moments have been determined correctly. Only the last cycle needs to be checked so the checking is easier. The convergence is generally rapid. It leads to the final solutions through just a small number of iteration cycles [27].

In Hardy cross method, end moments of the structural members, the unknowns, are obtained through iterating on their changes, whereas in Kani's Method, the iterations are performed on the unknowns themselves [27].

3.1.4.2 Research in Kani's Method and Moment Distribution method

Behravesh and Kaveh [26] described the relationship between Kani's and Hardy Cross approaches and a numerical iterative technique. They showed that the calculation trends— in both techniques—are like the Jacobi iterative procedure. A study of Volokh [24] also shows the correlation of Hardy Cross technique to the Jacobi iterative approach.

P. R. Patil, M. D. Pidurkar, and R. H. Mohankar [27] provided a comparison between both Kani's Method and Moment Distribution method through the analysis of a 2D single-bay portal frame for the case of vertical loading only. They deduced the following:

- The Final End Moments calculated by Kani's Method, for the considered portal frame, generally match with those calculated by Moment Distribution method.
- Only 3-4 iterations are enough while using Kani's Method, that is somewhat a relatively small number of iterations compared to that of Moment Distribution method, for the considered portal frame.
- Kani's Method is self-correcting.

A.S. Agrawal and U.S. Badgire [28] solved a 2D rigid jointed portal frame through using a simplified approach rather than the tedious calculations of the conversion factor and displacement factor. They separated non sway and sway analyses and calculated the final moments using ratio of the sway force and arbitrarily assigned sway force. The reason for their study is that the analysis of a portal frame that is rigidly jointed contains tedious calculations and much complication by using conversion factor and displacement factor in Kani's Method as displacement factor is required during the sway analysis. Their study showed that the incorporation of the simplified approach technique in Kani's Method makes it easier and quicker and decreases the tedious calculations since there is no need for the displacement factor and the conversion factor. Moreover, they mentioned that Simplified Approach technique is the same for the case of asymmetry in column height, asymmetric moment of inertia, and asymmetric support condition.

3.1.5 A new Jacobi-based iterative method for classical analysis of structures

Seyed M. Mirfallah and M. Bozorgnasab [10] provided a new technique for the classical computing. They named their proposed approach "Slope Distribution Method" (SDM). The SDM is an iterative technique that is based on successive computational cycles which are repeated till convergence.

SDM is based on Jacobi iterative procedure to find the values of the unknowns in the equations' system that is produced by the slope-deflection technique where the effects of shear and axial deformations are neglected, as their effect is relatively small compared to that of bending deformation. In SDM, the structural deformation values are attained without forming or solving the linear equations system which is its merit compared to the slope-deflection technique.

In contrast to both the Moment Distribution technique and Kani's Method, the distribution as well as the carry-over procedures are merged, thus only nodal slopes (rotations) are distributed rather than distributing and transmitting the bending moments at different members' ends that are attached to each rigid node. Accordingly, the analysis parameters and analysis time are reduced as the number of unknowns is dependent on the number of nodes of the structure and not on the number of members that are connected to each node. Consequently, the proposed procedure is less time-consuming than the methods of slope-deflection, Moment Distribution, and Kani's Method.

It should be noted that the sway displacement of a member is included within the proposed method in the form of sway rotation (lateral rotation) of the member. Finally, by obtaining the values of lateral rotation and nodal rotation of the story, the end moments and shear forces of the members can be calculated.

Additionally, they extended it to a matrix formulation in order to make the technique applicable and usable in computer software as only a matrix equation for unknowns is used.Moreover, they mentioned some special cases that SDM can analyze; these cases are:

- Frames with inclined columns.
- Frames with nodal vertical sway displacement.
- Structures that contain any non-prismatic member, however by defining some basic parameters (corresponding coefficients).
- Dual lateral load resisting systems. In other words, moment-resisting frames that contain other lateral load resisting elements, for instance, bracings. By applying some modifications, the lateral stiffness of the bracing members could be included in SDM equations.

3.2 Modern iterative (Krylov subspace) methods

3.2.1 Introduction and historical background

Krylov subspace techniques have undeniably become a popular and useful tool that is used for solving large groups of linear equations and nonlinear equations and finding eigenvalues, generalized eigenvalues besides singular values of matrix problems of large scale. Their generality is one of the explanations for their popularity. They are based on processes of projection onto Krylov subspaces. The approximations to solution are formed by minimizing residuals over the formed subspace.

The idea is named after naval engineer and Russian applied mathematician Alexei Nikolaevich Krylov (1863-1945), who published a paper in 1931 about it.

3.2.2 Iterative methods within this class

Among the well known Krylov subspace iterative methods are:

- Arnoldi: it is for solving eigenvalue problems.
- Lanczos: it is for solving eigenvalue problems.
- Conjugate Gradient (CG): it is the prototypical technique in this class; it is used for solving a system whose coefficient matrix [A] is symmetric positive-definite.
- Induced Dimension Reduction (IDR): it was presented originally to solve linear equations systems.
- Minimal Residual (MINRES): it is used in the case of symmetric and possibly indefinite matrices.
- Generalized Minimum Residual (GMRES): it is used in the case of non-symmetric matrices.
- Biconjugate Gradient (BiCG): it is used in the case of systems which are non-symmetric and/or indefinite.
- Biconjugate Gradient Stabilized (BiCGSTAB): it is used in the case of non-symmetric matrices. It is akin to the Conjugate Gradient Squared (CGS). The advantage of the BiCGSTAB is its limited storage needs, yet there are several problems for which BiCGSTAB technique does not work well. For these problems, GMRES method has become a better choice.
- The Conjugate Gradient Squared (CGS): it is a development of BiCG technique. It is an iterative technique for solving non-symmetric systems of linear equations.
- Quasi Minimal Residual (QMR): it is an iterative technique for solving non-symmetric systems of linear equations. QMR uses look-ahead procedures to avoid breakdowns within the underlying Lanczos process, that makes it more robust compared to BiCG.
- Transpose-free QMR (TFQMR): it is an iterative technique for solving non-symmetric systems of linear equations. Conceptually, it is derived from CGS method.When the CGS technique shows irregular convergence, the TFQMR technique can produce much smoother and almost monotonic convergence curves. Nevertheless, the close relationship between CGS and TFQMR technique suggests that the overall convergence speed is similar for both approaches. However, the TFQMR method, in some cases, may converge faster than CGS method.

3.2.3 Conjugate Gradient (CG) method

This section is designated to shed light on the CG method as it is a well-known Krylov subspace method and a prototypical technique of this class.

3.2.3.1 Introduction

The Conjugate Gradient iterative method (CG) was originally devised by Hestenes and Stiefel in 1952 [29]. As mentioned previously, it is an iterative technique for solving a system of linear equations whose coefficient matrix [A] is a symmetric positive definite matrix.

As mentioned in [30], a quadratic form is a quadratic, scalar function of a vector by the form of (6):

$$f(x) = \frac{1}{2} x^T Ax - b^T x + c \quad (6)$$

Where A is a matrix of nxn order, x and b are n-vectors, and c is a constant. If A is a symmetric, positive-definite matrix, $f(x)$ is minimized through the solution to $Ax = b$.

The Conjugate Gradient is the archetype of Krylov space solvers which is an orthogonal projection technique and satisfies a condition of minimality. In this method, the step taken along each direction is chosen in order to minimize a function which measures the residual along that direction. The major feature of CG is that its search directions are mutually conjugate, and hence the finite termination property is given[31].

The rate of convergence of CG technique is known to be reliant on the eigenvalues' distribution of the coefficient matrix [A]. However, CG method does not require previous knowledge of the eigenvalues; it implicitly takes into account the eigenvalues' distribution. Thus, estimation of over-relaxation parameter is not needed[32].

Conjugate Gradient methods are also used for solving nonlinear equations and unconstrained optimization, especially in large-scale cases. As they have the attractive practical factors concerning low memory requirement and simple computation as well as interesting theoretical features of the curvature information and also strong global convergence [33].

In fact, the CG technique is not among the most robust or fastest optimization algorithms available today for nonlinear problems, however, it remains popular for mathematicians and engineers who are keen on solving large problems [34].

3.2.3.2 Historical background and literature review

As mentioned in [31],[32]; Reid [35] ignited a lot of the subsequent development of conjugate gradient approaches for solving algebraic equations' systems; he showed that Conjugate Gradient method is a powerful iterative procedure for suitable systems. Moreover, the Conjugate Gradient technique was applied by Meijerink and Vorst [36] in conjunction with the incomplete Cholesky decomposition for M-matrices. They explained how to adjust a general system, for example, those derived from models of finite-element or finite-difference, into a 'desirable' form which Conjugate Gradient could efficiently be applied to. This was achieved through preconditioning the equations system with an easily generated approximate coefficient matrix inverse. Since then, some more effective preconditioning techniques have been developed.

A history and an extensive bibliography of CG method to the mid-seventies are given by Golub and O'Leary [37]. Since then, most research has focused on nonsymmetric systems[30]. Barrett et al. [38] offered a survey of iterative methods for the solution of linear systems.

A strongly implicit pre-conditioned CG procedure's form was considered by P. K. Khosla and S. G. Rubin [32]. The resulting iterative method was applicable to the sparse systems of difference equations that arise from boundary value problems. Moreover, quasi-Newton methods were introduced to accelerate the convergence rate of strongly implicit finite-difference iterative methods. The resulting procedures were quite fast and could easily be programmed.

D.A.H. Jacobs [31] described a generalization to complex systems by developing Fletcher's work [39]. The Methods were improved on 'symmetrization', solving normal equations for asymmetric cases, besides expanding complex systems to real systems of twice the order. Jacobs has shown and proved the extension of biCG technique to complex systems of equations, hence the new method was called complex biconjugate Gradient. The author proved that the Method was effective.

Shengwei Yao, Xiwen Lu, and Zengxin Wei [34] proposed a CG technique which resembles Dai-Liao Conjugate Gradient technique [40] but had relatively stronger convergence properties. The proposed technique owns the sufficient descent condition. It is also globally convergent under the line search of strong Wolfe-Powell (SWP) for general function. The provided numerical results showed that the proposed technique was remarkably efficient for the test problems.

Can Li [41] further studied the CG technique for unconstrained optimization and focused his attention on the descent CG technique. The modified Conjugate Gradient method that has been presented had an interesting feature since the direction was always a descent one for the objective function. Additionally, the property was independent of the used line search. The author proved that, under mild conditions, the modified CG technique with Armijo-type line search was globally convergent. He also presented some numerical results that showed the efficiency of the proposed technique.

XiaoPing Wu, LiYing Liu, FengJie Xie, and YongFei Li [42] proposed a new nonlinear CG formula, that satisfies the condition of sufficient descent, for solving unconstrained optimization problems. The algorithm's global convergence was implemented under weak Wolfe line search. This new algorithm was shown to be competitive with other earlier algorithms by some numerical experiments.

Xiangrong Li, Xiaoliang Wang, Zhou Sheng, and Xiabin Duan [33] presented a modified CG algorithm by line search technique with acceleration scheme for the case of nonlinear symmetric equations. Moreover, the proposed technique not only owns descent property but also possesses global convergence within mild

conditions. Numerical results also show that the presented technique is much more effective compared to the other approaches for the test problems.

3.2.3.3 Ill-conditioning and round-off error

The Conjugate Gradient reaches the exact solution after a number of n iterations. Provided that there is infinite floating point precision, the required number of iterations to yield an exact solution will be the number of distinct eigenvalues at most[30], [43]. Since the available number of digits is limited, round-off errors occur through each numerical operation. Digits are lost when small numbers are added to large numbers, and this may be caused via large differences within the scale of unknowns. Accordingly, orthogonality among the computed vectors is usually lost very quickly, accompanied by a consequent loss of linear independence.

In practice, the accumulated floating point round-off error can delay the convergence, as the solver may take more than n steps, or even fail to converge occasionally; this is as a result of gradually losing the residuals' accuracy. This effect can be alleviated by periodically using the equation $r_i=b-Ax_i$ to recalculate the correct residuals[30].

Nonetheless, a poorly conditioned coefficient matrix [A] is the main cause of rounding error. Therefore, Preconditioning is the solution to this round-off error[43].

3.2.3.4 Preconditioning

Generally, it is accepted that CG technique should almost always be used with an appropriate preconditioner, especially for large-scale applications. Several preconditioners have been developed, yet some of these preconditioners are quite sophisticated[30].

The general concept underlying any preconditioning approach for the iterative solvers is to alter the ill-conditioned system which is $A\ x = b$, whose condition number is high, in such a way in order to obtain an equivalent system that is $\widehat{A}\,\widehat{x} = \widehat{b}$ for which the iterative technique converges faster as the system's condition number is improved.

There is a standard approach that is to use a matrix [M] which is nonsingular, then rewrite the system as $M^{-1} A\ x = M^{-1}\ b$. The preconditioner matrix [M] needs to be selected such that the modified coefficient matrix, $\widehat{A}=M^{-1}A$, will be better conditioned for the CG technique compared to the original coefficient matrix [A].

A perfect preconditioner will be $M=A$ where $M^{-1}A$ is equal to the identity matrix. However, the preconditioning step is, unfortunately, solving the system $M\ x = b$, accordingly this isn't a meaningful preconditioner at all[30]. On the contrary, the least effective preconditioner $M = I$ which absolutely does nothing. Hence, in practice, it should be tried to get a preconditioner which is something in between[44].

Finding a convenient way of doing preconditioning may be sometimes a difficult task, however, it can likewise result in significantly impressive convergence results. It should also be taken into consideration that, in numerous times, conducting experiments is the only real methodology of determining which way of preconditioning works best[44].

IV. Conclusion

Direct solvers generally tend to need remarkably large storage demands and a great amount of calculation, especially for huge problems. Thus, it spends a long run time. In these cases, iterative solvers that require relatively less memory space, are more desirable; they are also faster than direct solvers in solving such huge problems. Although the single iteration of the classical iterative methods generally takes a shorter time than that of the modern iterative methods, as classical methods have simpler steps (algorithms), the modern methods are generally faster as they take a fewer number of iterations till convergence. However, the classical methods are simpler to program and when they are combined with the modern methods, e.g., as preconditioners, they can be quite successful. Finally, it should be noted that the research in this field is always needed to attain new faster solvers or even improving the available methods by some modifications to overcome their weak points as there is no method that is the most suitable to be used in all cases and conditions.

References

[1] A. Harish, "How to Choose a Solver for FEM Problems: Direct or Iterative?," *SIMSCALE Blog*, 2018. [Online]. Available: https://www.simscale.com/blog/2016/08/how-to-choose-solvers-for-fem/. [Accessed: 09-May-2019].

[2] W. Frei, "Solutions to Linear Systems of Equations: Direct and Iterative Solvers," *COMSOL Blog*, 2013. [Online]. Available: https://www.comsol.com/blogs/solutions-linear-systems-equations-direct-iterative-solvers/. [Accessed: 09-May-2019].

[3] Cyprien, "What is an FEA Solver ?," *FEA forAll*, 2013. [Online]. Available: http://feaforall.com/what-is-an-fea-solver/. [Accessed: 09-May-2019].

[4] A. H. Laskar and S. Behera, "Refinement of Iterative Methods for the Solution of System of Linear Equations Ax=b," *IOSR J. Math.*, vol. 10, no. 3, pp. 70–73, 2014.

[5] AUTODESK.Help, "Solvers in Finite Element Analysis," *AUTODESK KNOWLEDGE NETWORK-SIMULATION MECHANICAL*, 2017. [Online]. Available: https://knowledge.autodesk.com/support/simulation-mechanical/learn-explore/caas/CloudHelp/cloudhelp/2018/ENU/SimMech-UsersGuide/files/GUID-8037D314-5AAE-47E0-8B42-3C91EEFBA15D-htm.html. [Accessed: 09-May-2019].

[6] H. M. Antia, *Numerical methods for scientists and engineers*. Hindustan Book Agency, 2012.

[7] A. D. Tuff and A. Jennings, "An iterative method for large systems of linear structural equations," *Int. J. Numer. Methods Eng.*, vol. 7, no. 2, pp. 175–183, 1973.

[8] Wikipedia, "Condition number," *Wikimedia Foundation*, 2020. [Online]. Available: https://en.wikipedia.org/wiki/Condition_number. [Accessed: 25-Sep-2020].

[9] A. Bakari and I. Dahiru, "Comparison of Jacobi and Gauss-Seidel Iterative Methods for the Solution of Systems of Linear Equations," *Asian Res. J. Math.*, vol. 8, no. 3, pp. 1–7, 2018.

[10] S. M. H. Mirfallah and M. Bozorgnasab, "A New Jacobi-based Iterative Method for the Classical Analysis of Structures," *Lat. Am. J. Solids Struct.*, vol. 12, pp. 2581–2617, 2015.

[11] H. Liu and T. Feng, "Study on the convergence of solving linear equations by gauss-seidel and jacobi method," *Proc. - 2015 11th Int. Conf. Comput. Intell. Secur. CIS 2015*, pp. 100–103, 2016.

[12] Z. L. Zhang Jing, "Comparison of the convergence rate in iterative methods," *J. Bohai Univ. Nat. Sci. Ed.*, vol. 2, pp. 163–165, 2007.

[13] D. K. Salkuyeh, "Generalized Jacobi and Gauss-Seidel Methods for Solving Linear System of Equations," vol. 16, no. 2, pp. 164–170, 2007.

[14] M. H. Schultz, "Gmres: a generalized minimal residual algorithm for solving nonsymmetric linear," no. 3, pp. 856–869, 1986.

[15] van der V. H. A., "Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 13, no. 2, pp. 631–644, 1992.

[16] S. Y., *Iterative Methods for Sparse Linear Systems*. New York: PWS Press, 1995.

[17] B. N. Datta, *Numerical Linear Algebra and Applications*. Brooks/Cole Publishing Company, 1995.

[18] S. P. Frankel, "Convergence Rates of Iterative Treatments of Partial Differential Equations," *Math. Tables Other Aids to Comput.*, vol. 4, no. 30, pp. 65–75, Jul. 1950.

[19] D. M. Young, "Iterative methods for solving partial differential equations of elliptic type," Harvard University, Cambridge, MA, 1950.

[20] R. C. and M. Wei, "Research and Application of Successive Over-Relaxation Iterative Algorithm," *2011 Int. Conf. Electron. Mech. Eng. Inf. Technol.*, pp. 3856–3858, 2011.

[21] T. Mayooran and E. Light, "Applying the Successive Over-relaxation Method to a Real World Problems," *Am. J. Appl. Math. Stat.*, vol. 4, no. 4, pp. 113–117, 2016.

[22] A. Hadjidimos, "Accelerated Overrelaxation Method," *Math. Comput.*, vol. 32, no. 141, pp. 149–157, 1978.

[23] H. Cross, "Analysis of Continuous Frames By Distributing Fixed-End Moments," *Proc. Am. Soc. Civ. Eng.*, pp. 919–928, 1930.

[24] K. Volokh, "On foundations of the Hardy Cross method," *Int. J. Solids Struct.*, vol. 39, pp. 4197–4200, 2002.

[25] G. Kani, *Analysis of multistory frames*. Burns & Oates, 1957.

[26] A. Behravesh, A., and Kaveh, "Iterative solution of large structures," *Comput. Struct.*, vol. 35, pp. 279–282, 1990.

[27] R. H. M. P. R. Patil, M. D. Pidurkar, "Comparative Study of End Moments Regarding Application of Rotation Contribution Method (Kani's Method) & Moment Distribution Method for the Analysis of Portal Frame," *IOSR J. Mech. Civ. Eng.*, vol. 7, no. 1, pp. 20–25, 2013.

[28] A. S. Agrawal and U. S. Badgire, "Sway Analysis of Rigid Jointed Portal Frame by Using Simplified Approach Method in Rotation Contribution Method (Kani ' s Method)," vol. 7, no. 8, pp. 64–69, 2017.

[29] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *J. Res. Natl. Bur. Stand. (1934).*, vol. 49, no. 6, pp. 409–436, 1952.

[30] J. R. Shewchuk, "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain." School of Computer Science, Carnegie Mellon University, Pittsburgh, 1994.

[31] D. A. H. Jacobs, "A generalization of the conjugate-gradient method to solve complex systems," *IMA J. Numer. Anal.*, vol. 6, no. 4, pp. 447–452, 1986.

[32] P. K. Khosla and S. G. Rubin, "A conjugate gradient iterative method," *Comput. Fluids*, vol. 9, no. 2, pp. 109–121, 1981.

[33] X. Li, X. Wang, Z. Sheng, and X. Duan, "A modified conjugate gradient algorithm with backtracking line search technique for large-scale nonlinear equations," *Int. J. Comput. Math.*, vol. 95, no. 2, pp. 382–395, 2017.

[34] S. Yao, X. Lu, and Z. Wei, "A conjugate gradient method with global convergence for large-scale unconstrained optimization problems," *J. Appl. Math.*, vol. 2013, no. 3, 2013.

[35] J. K. Reid, *On the method of conjugate gradients for the solution of large sparse systems of linear equations. In: Large Sparse Sets of Linear Equation*. Academic Press, London, 1971.

[36] J. A. Meijerink and H. A. van der Vorst, "An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix is a Symmetric M-Matrix," *Math. Comput.*, vol. 31, no. 137, pp. 148–162, 1977.

[37] G. H. G. and D. P. O'Leary, "Some History of the Conjugate Gradient and Lanczos Algorithms: 1948–1976," *SIAM Rev.*, vol. 31, no. 1, pp. 50–102, 1989.

[38] V. Richard Barrett, Michael Berry, Tony Chan, James Demmel, June Donato, Jack Dongarra and and H. van der V. Eijkhout, Roldan Pozo, Charles Romine, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. Philadelphia, Pennsylvania: SIAM, 1993.

[39] R. Fletcher, "Conjugate gradient methods for indefinite systems," in *Numerical Analysis*, 1976, pp. 73–89.

[40] Y. H. Dai and L. Z. Liao, "New conjugacy conditions and related nonlinear conjugate gradient methods," *Appl. Math. Optim.*, vol. 43, no. 1, pp. 87–101, 2001.

[41] C. Li, "A Modified Conjugate Gradient Method for Unconstrained Optimization," *TELKOMNIKA*, vol. 11, no. 11, pp. 6373–7380, 2013.

[42] X. Wu, L. Liu, F. Xie, and Y. Li, "A New Conjugate Gradient Algorithm with Sufficient Descent Property for Unconstrained Optimization," *Math. Probl. Eng.*, vol. 2015, no. 7, 2015.

[43] M. Rambo and H. de Moor, "The Conjugate Gradient Method for Solving Linear Systems of Equations." Department of Mathematics, Saint Mary's College of California, Moraga, California, 2016.

[44] R. H. Refsnæs, "A BRIEF INTRODUCTION TO THE CONJUGATE GRADIENT METHOD." 2009.

Baher A. Haleem, et. al. "An Overview of Iterative Techniques Used for Solving the Systems of Equations Resulting from the Finite Element Analysis." *IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE)*, 17(5), 2020, pp. 48-57.