ASIAN JOURNAL OF CIVIL ENGINEERING (BHRC) VOL. 15, NO. 4 (2014)
PAGES 547-561

PRECONDITIONED IMPROVED BI-CONJUGATE GRADIENT IN NONLINEAR ANALYSIS OF SPACE TRUSSES

H. Saffari[1*], A. Maghami[1] and I. Mansouri[2]
[1]Department of Civil Engineering, Shahid Bahonar University of Kerman, P.O. Box 76169133, Kerman, Iran
[2]School of Civil Engineering, Birjand University of Technology

Received: 2 January 2014; **Accepted:** 24 March 2014

ABSTRACT

Different methods for analyzing nonlinear systems have been proposed. Most of them are not fast enough and also fail to deal with particular set of equations like nonlinear analysis of structures. In this paper, the preconditioned biconjugate gradient (PBCG) is combined with the Newton-Raphson method to reduce the convergence time of each iteration in nonlinear solution process. This algorithm replaces the procedure of inversing the tangent stiffness matrix with a new iterative method to solve the linearized system of equations. Also a new method is introduced to reduce computational time by applying preconditioning to the improved biconjugate gradient method. The proposed method is useful to analyze complex behavior structures, including unloading, snap-through buckling, and inelastic post-buckling analyses. In this paper a new preconditioner is proposed which is suitable for nonlinear analysis of structures. Results show that using the new methodology maintains the accuracy in an acceptable level while reduces computational cost and time, significantly.

Keywords: Nonlinear analysis; space truss; iterative methods; biconjugat gradient; preconditioning.

1. INTRODUCTION

Recently, many practicing engineers tend to analyze nonlinear structures; in addition the linear analysis of structures has become insufficient for most structural engineering applications. So in the last decades, the analysis of non-linear problems became a noticeable subject. One powerful approach to evaluate the response of a structure to a set of successive loads is the Newton-Raphson method. When the solution is close to a limit point, this method diverges [1].A limit point describes the turning point for the equilibrium path of a

* E-mail address of the corresponding author: hsaffari@uk.ac.ir (H. Saffari)

structure, which can be further regarded as the transition point from stable to unstable equilibrium states or vice versa [2]. Many solution algorithms and combined methods have been presented to overcome this limitation. Kwasniewski [3] proposed the complete equilibrium paths for several Mises trusses. Riks [4] suggested the arc-length method, and Crisfield [5] proposed several different versions of that method by updating constraint equations. To study snap-through buckling problems Bellini [6] suggested a simple mathematical model. Ragon et al. [7] compared three algorithms for tracing nonlinear equilibrium paths of structural systems. Eriksson [8] introduced several path-following procedures, based on expressions from analytical elastic stability theory for geometrically nonlinear structural analysis. Several analytical techniques have been used by researchers to analyze the behavior of trusses. A modified normal flow algorithm has been applied by Saffari et al. [9] to study nonlinear behavior in truss structures. This method was applied to several applications [9, 10, 11]. The Newton-Raphson method advanced by many iterative methods [12, 13, 14]. Papadrakakis and Gantes [15] presented some procedures to shorten the Newton-Raphson method. Greco et al. [16] suggested a new geometric non-linear formulation for space truss analysis that works with nodal positions rather than nodal displacements. The normal flow algorithm was also used by Tabatabaei and Saffari [10] to study large strain analysis of planer frames. Thai and Kim [17] introduced the large-deflection inelastic analysis of space trusses using both geometric and material non-linearities. An amazing procedure for nonlinear analysis of structures presented by Saffari and Mansouri [18] which accelerates the convergence rate. Saffari and Mansouri[18]used a mathematical method which was named the two-point method, to attain the convergence state. Saffari et al. [19] also applied an algorithm namely three-point enables to pass the limit points with low computational cost. Researchers presented various methods of enhancement of analysis. Cadou et al. [20] evaluated two convergence accelerating techniques known as the modified minimal polynomial extrapolation method (MMPE) and the Pade approximants. The homotopy perturbation method was proposed to reduce the computational time for elasto-plastic analysis of frames by Saffari et al. [21].Saffari et al. [22] used an eight-order convergence method to achieve faster convergence. In most solution methods which are described earlier, the efficiency of methods developed were dominated by their mathematical features.

At the present time, the conjugate gradient (CG) method is a well-known and favored iterative method for solution of large systems of equations [23, 24, 25]. One of amazing aspect is that the method reaches a satisfactory accuracy even if the size of the matrix is large [26].

Papadrakakis and Gantes [15] employed procedures for the solution of large-scale nonlinear equations. They employed a conjugate gradient method in the solution procedure, which was halted at the limit point though.

Saffari et al. [27] combined the Newton-Raphson method with three algorithms and concluded that biconjugte gradient (BCG) method is a more effective scheme to be coupled with the Newton-Raphson method in nonlinear analyses of structures. They also improved BCG method for solving a system of nonlinear equations which reduces the computational time and the number of iterations.

The BCG is an iterative method which is used widely in science and engineering problems. The performance of this method can be dramatically increased if it is applied with

a suitable preconditioner [28].

Usually, iterative methods are applied with a preconditioner (M)—so, they are applied to the system $Max = Mb$, where M is a matrix which approximates in some sense the inverse of A, and M should be easy to compute [29].

This paper is concerned with a new approach using preconditioning for solving linear part of the problem. The preconditioning technique is applied to the improved BCG which can directly solve the nonlinear system of equations. Our goal is to develop a suitable preconditioned algorithm for the nonlinear analysis of structures. A computer program was presented that incorporates the PBCG method and the Preconditioned Improved BCG method using three kinds of preconditioner.

2. NONLINEAR ANALYSIS OF TRUSSES

2.1 Geometrically nonlinear analysis

Elementary information can be found in [30].

The equations of equilibrium of system can be shown as

$$\{f(\mathrm{u})\} = \{P\} \tag{1}$$

The resultant of the nodal internal forces is $\{f(u)\}$ and $\{P\}$ represents the external nodal loads. The member force deformation relationships indicate that $\{f\}$ is a highly non-linear function of u .

In accordance with load-deflection relations, it is rarely possible to explicitly solve these equations. The differential form of above equation is

$$[\tau]\{\Delta u\} = \{\Delta P\} \tag{2}$$

In which, $\{\Delta u\}$, $\{\Delta P\}$ and $[\tau]$ stands for increments, load and the system tangent stiffness matrix respectively.

2.2 Material nonlinearity analysis

The exactitude in the structure inelastic response depends on the accuracy of the member's load–displacement relationship used in the analysis. To predict the nonlinear behavior of space trusses a number of models have been introduced. In this paper, a stress–strain relationship was suggested by Hill et al. [31] which is adopted to predict the inelastic post–buckling behavior of the trusses.

To calculate the internal force of the truss member in local coordinates the axial deformation is used. It is denoted by u , such that $u = L_2 - L_1$

Where L_1 and L_2 is the length of the member before and after deformation.

In elastic case:

$$Q = \frac{AE}{L} u \tag{3}$$

In inelastic case:
- For tensile members:

$$Q = \begin{cases} \frac{AE}{L} u & for \quad u < u_y \\ AF_y & for \quad u \geq u_y \end{cases} \tag{4}$$

Where F_y denotes yield stress and $u_y = F_y L / E$.
- For compressive members:

$$Q = \begin{cases} \frac{AE}{L} u & for \quad |u| < u_y \\ Q_l + (Q_{cr} - Q_l).e^{\left[-(X_1 + X_2\sqrt{u'/L})u'/L\right]} & for \quad |u| \geq u_y \end{cases} \tag{5}$$

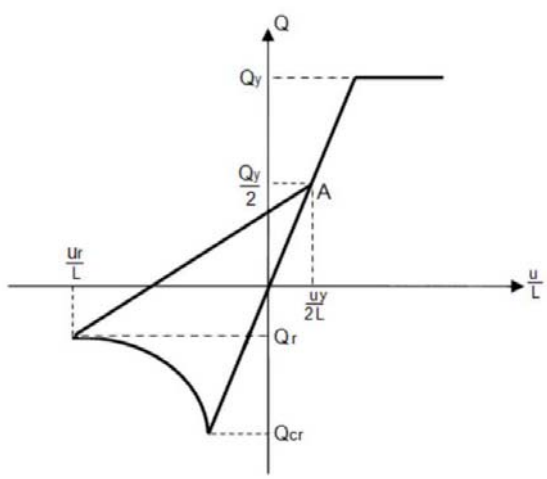

Figure 1. Assumed force-deformation curve for loading and unloading path

Here $Q_{cr} = \pi^2 EI / L^2$ (I = weak axis moment of inertia) and Q_l is the asymptotic lower stress limit defined as $Q_l = r.Q_{cr}$. $u_{cr} = Q_{cr} L/(AE)$ is the corresponding critical buckling displacement. u' is defined as $u' = u - u_{cr}$. Parameters X_1 and X_2 are constants and depends on the slenderness ratio of the compressive members.
When a member is in compression state and $u \geq u_{cr}$ for the inelastic case, the tangent modulus, E_t, has to be used in place of E. The tangent modulus is obtained by:

$$E_t = -\frac{1}{A}(Q_{cr} - Q_l).e^{\left[-(X_1 + X_2\sqrt{u'/l})u'/L\right]}.(X_1 + \frac{3}{2} X_2 \sqrt{u'/L} \tag{6}$$

The unloading path of the buckled members in the inelastic range is shown in Figure 1.So, if

loading path reaches point A, the member behavior follows relations in Eq. ((4).

3. PRECONDITIONING

In spite of essential request of iterative techniques in industrial applications for solving large linear systems, these methods simply rejected due to their less robustness than direct solvers. Preconditioning improves both the robustness and efficiency of iterative methods [28]. Preconditioning is a means of transforming the system of equations in to another system with more favorable properties for iterative solution. A preconditioner is a matrix that effects such a transformation [32].

The idea behind preconditioning is to replace $Ax = b$ by

$$M^{-1}Ax = M^{-1}b$$

Or

$$AM^{-1}u = b\,,\ \ x = M^{-1}u$$

A matrix which approximates A in some sense is the preconditioner M. The preconditioned algorithms all need a linear system solution with the matrix M at each step so the preconditioner (M) should be chosen in a way which is inexpensive to solve linear systems $Mx = b$ [28].

A good preconditioner M should have these properties:

1) The preconditioned system should be easy to solve which means the preconditioned iteration should converge rapidly

2) The preconditioner should be cheap to construct and apply.

These two requirements are in competition with each other. It necessitates to strike a balance between the two needs [32].

Finding a good preconditioner to solve a given sparse linear system is a combination of art and science. Theoretical results are rare but in spite of expectations some methods work surprisingly well. The Jacobi preconditioner is one of the simplest forms of preconditioning, in which the preconditioner is chosen to be the diagonal of the matrix A. There are no limits to available options for constructing good preconditioners. For example, preconditioners can be derived from knowledge of the original physical problems from which the system of equations arises [28].

When the PBCG and N-R methods are combined, in each iteration of the N-R method a new preconditioner is used. Since the tangent stiffness matrix often changes gradually along the equilibrium path in nonlinear analysis of structures, a preconditioner can be used in more than one iteration of the N-R method. In this paper, the inverse of the first tangent stiffness matrix is used as a preconditioner of the other iterations. In other words, the inversing of tangent stiffness matrix is used only at the first iteration which does not need significant computing time. Applying this method results in less computing time comparied to the BCG method. If the tangent stiffness matrix changes significantly in the path of equilibrium, the iteration of PBCG method increases, thus a preconditioner should be upgraded and re-

applied to the system of equations. Therefore If the upgrading the preconditioner is required, the inverse of tangent stiffness matrix can be used as a new preconditioner.

4. PRECONDITIONED IMPROVED BCG ALGORITHM

The PBCG algorithm can be explain in a step-by-step procedure as follows:

1. Choose an initial guess x_0; compute $r_0 = b - Ax_0$ and $r_0{}^* = b^* - x_0{}^* A^T$; choose a preconditioner M

2. Set $p_0 = M^{-1} r_0$ and $p_0{}^* = r_0^* M^{-1}$

3. For $i = 1,2,3,...$ calculate the following in stepwise manner:

a. $\alpha_i = \frac{r_i^* M^{-1} r_i}{p_i^* A p_i}$

b. $x_{i+1} = x_i + \alpha_i p_i$

c. $r_{i+1} = r_i - \alpha_i A p_i$

d. $r_{i+1}^* = r_i^* - \alpha_i p_i^* A^T$

e. $\beta_i = \frac{r_{i+1}^* M^{-1} r_{i+1}}{r_i^* M^{-1} r_i}$

f. $p_{i+1} = M^{-1} r_{i+1} + \beta_i p_i$

g. $p_{i+1}^* = r_{i+1}^* M^{-1} + \beta_i p_i^*$

h. Repeat steps a-g until convergence is reached; i.e., if x_{i+1} is close enough to x_i, stop-otherwise, return to step 3

Mr and r^*_ should not be orthogonal. If there is no preconditioning, then r and r^*_ should not be orthogonal (van der Vorst 2003).

Saffari et al. [27] improved the BCG method by changing it that instead of seeking convergence on a tangent path, the equilibrium path is followed until convergence is reached. Classic BCG and improved BCG approaches are schematically shown in Figure 2. Figure 2shows how the BCG method was improved in such a way that directly solves the nonlinear system of equations and the preconditioned shape of this method is that we use:

$$r_{i+1} = r_i - f(\delta_{i+1}) \tag{7}$$

Instead of step 3-c, the algorithm is improved (f = resultant internal forces).

An iterative method uses an initial guess to generate successive approximations to a solution. The PBCG method has primarily shown that it needs an initial guess (x_0) in order to start the process, so in the second iteration of the N-R, the answer of the first iteration is achieved. The answer is usually close to the answer of second iteration. If the answer of the prior iteration is used as initial guess (x_0) for the next iteration, then the equations can be solved by less number of iterations.

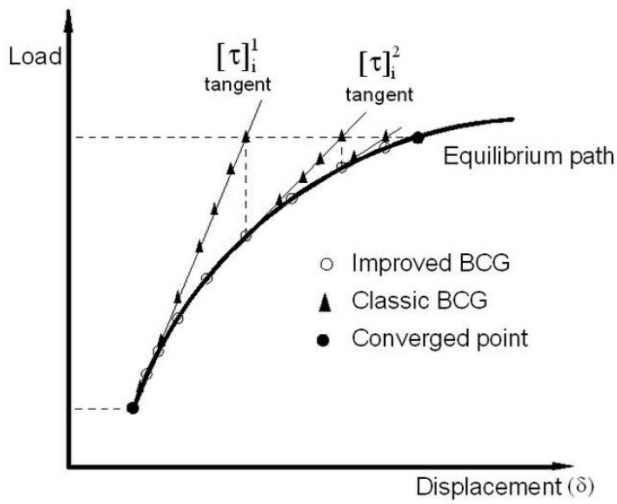

Figure 2. BCG algorithm approach [27]

5. NORMAL FLOW ALGORITHM

As shown in Figure 3 the normal flow algorithm can accurately follow the equilibrium path of nonlinear problems with multiple limit points and snap-through points.
$\{P\}_i^j$ is the total load on the structure, the i is the step number and j is the number of modifying iteration. Mathematically, this is written as

$$\{P\}_i^j = \lambda_{toti}^j \{P_{ref}\} \tag{8}$$

In which, λ_{toti}^j is the factor of total external load and $\{P_{ref}\}$ is the reference external load vector.
A detailed information and the process of the normal flow algorithm have been represented in reference [9]
ψ is the vector of the unbalanced forces and a particular solution $\{V\}$ should be obtained from:

$$[\tau]_i^{j-1}\{V\} = \Delta\lambda_i^j \{P_{ref}\} - \{\psi\}_i^{j-1} \tag{9}$$

In which:

$$\Delta\lambda_i^j = -\frac{\left[\{u_I\}_i^j\right]^T \{\Delta u_R\}_i^j}{\left[\{u_I\}_i^j\right]^T \{u_I\}_i^j} \tag{10}$$

$\{\Delta u_R\}_i^j$ is the vector of unbalanced displacements, $\{u_I\}_i^j$ is vector of tangential displacement in the converged point.

$$\{\psi\}_i^{j-1} = \{F_{\text{int}}\}_i^j - \left(\lambda_{toti-1}^{conv} + \lambda_i^{j=1} + \sum_{j=2} \Delta\lambda_i^{j-1}\right)\{P_{ref}\} \tag{11}$$

$\{F_{\text{int}}\}_i^j$ is the vector of resultant internal force at the nodes. Using the following system equations the vector of unbalanced force is calculated:

$$[\tau]_i^{j-1}\{\Delta u_R\}_i^j = -\{\psi\}_i^{j-1} \tag{12}$$

The minimum solution of the norm is computed through the following equation:

$$\{\Delta u\}_i^j = \{V\} - \frac{\{V\}^T\{u_I\}_i^j}{\|u_I\|^2}\{u_I\}_i^j \tag{13}$$

Where $\{u_I\}_i^j$ is the tangential displacement. A direct method of updating is applied such that, the load increment is related to the number of iterations. Using the following relationship the sign of determinant of the tangential stiffness matrix of the previous step is computed.

$$\lambda_{i+1}^j = \pm\lambda_i^j\left(\frac{J_D}{J_M}\right)^{\gamma} \tag{14}$$

In which γ is a certain number, J_M is the number of iterations performed in the previous step and J_D is the number of iterations assumed at the beginning of the computations.

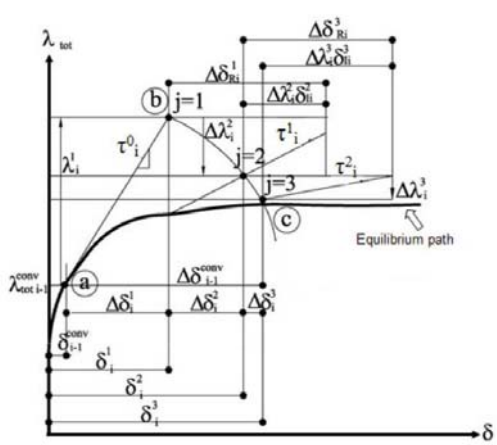

Figure 3. Normal flow algorithm scheme [27]

6. NUMERICAL RESULTS

In this study, numerical examples are presented and are solved using the Newton–Raphson method and four algorithm discussed earlier. It should be noted that the PBCG method and PIBCG method applied with 3 different preconditioners which was mentioned earlier. The program is written in MATLAB code.

In this article some examples are presented to examine the performance of the method proposed here and to evaluate the result sobtained. In order to solve the system of equations, the numerical algorithms are applied to two cases of elastic and inelastic post-buckling analyses of typical and well-known trusses. The CPU time taken by a calculation process can be measured. All examples are solved using a 32-bit Pentium 2.00 GHz processor (Dual core)

Example 1:

The geometry and loading of a geometric truss with 61 nodes and 156 elements are shown in Figure 4. All of elements have equal cross sections. The cross section of all members is $A = 6.5cm^2$ and their moment of inertia is $I = 1cm^4$. The elasticity modulus of the truss members is $E = 6895kN/cm^2$ and $F_y = 400\,kN/cm^2$ the geometric dimensions of this truss are adopted from Ramesh and Krishnamoorthy [33]. A concentrated vertical load of $P = 8kN$ is applied to the center of dome. The elevation of the space truss is defined by the following equation:

$$x^2 + y^2 + (z + 7.2)^2 = 60.84 \tag{15}$$

For Eq. (14) parameters $\Delta\lambda_1^1 = 0.01$, $\lambda_{\max} = 1$, $\gamma = 0.1$, $J_D = 2$, $J_{\max} = 100$ are adopted.

Figure 5 shows variation of vertical displacement at central node with the load P obtained in two cases of analysis.

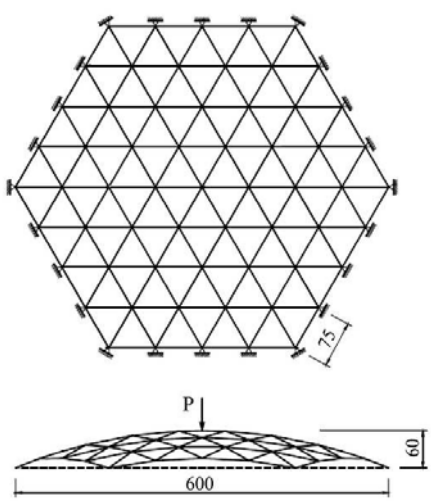

Figure 4. Geodesic dome truss, dimensions are given in cm.

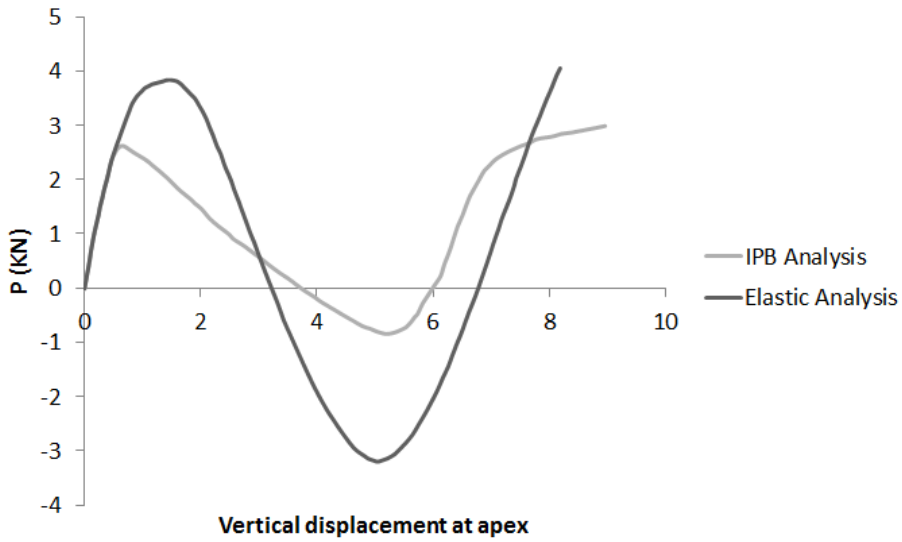

Figure 5. Load-displacement curves of geodesic dome truss at apex

Table 1 presents a summary of the CPU time taken for application of classic Newton-Raphson (N-R) and four different algorithms, including combination of Newton-Raphson and biconjugate gradient (N-R + BCG), combination of Newton-Raphson and Preconditioned biconjugate gradient (N-R + PBCG), Improved biconjugate gradient(IBCG), and Preconditioned improved biconjugate gradient(PIBCG), in two cases of analysis. It should be mentioned that the PIBCG method and PBCG method with three different preconditioners have been used. These preconditioners include The Jacobi preconditioner (diag) which is chosen to be the diagonal of the tangent stiffness matrix, the invers of first tangent matrix (p1) and a preconditioner which changes in some iteration (p2).

Table 1: Comparison of CPU Time (sec) for Example 1

		Analysis	
method	preconditioner	Elastic	IPB
N-R		27.5305	48.0764
N-R+BCG		12.3621	19.5231
	diag	11.3906	18.1975
NR+PBCG	p1	10.1498	15.8162
	p2	9.4622	14.7228
IBCG		9.0356	11.1679
	diag	8.8789	10.3214
PIBCG	p1	7.3568	9.1692
	p2	6.9965	8.9484

Example 2:

The circular dome truss with 168 elements and 73 nodes taken from Thai and Kim [17] is

shown in Figure 6. This space truss is subjected to a vertical load $P = 8kN$ at the apex.
The cross section of all members is $A = 50.431cm^2$ and their moment of inertia is $I = 52.94cm^4$. The elasticity modulus of the truss members is $E = 2.04 \times 10^4 kN/cm^2$ and $F_y = 25kN/cm^2$ For Eq. (14) parameters $\Delta\lambda_1^1 = 0.01$, $\lambda_{\max} = 2$, $\gamma = 0.1$, $J_D = 5$, $J_{\max} = 100b$ are adopted.
To compare the performance of five different algorithms, the results of analyses are summarized in Table 2. It can be easily seen that less computing time is used by PIBCG algorithm.

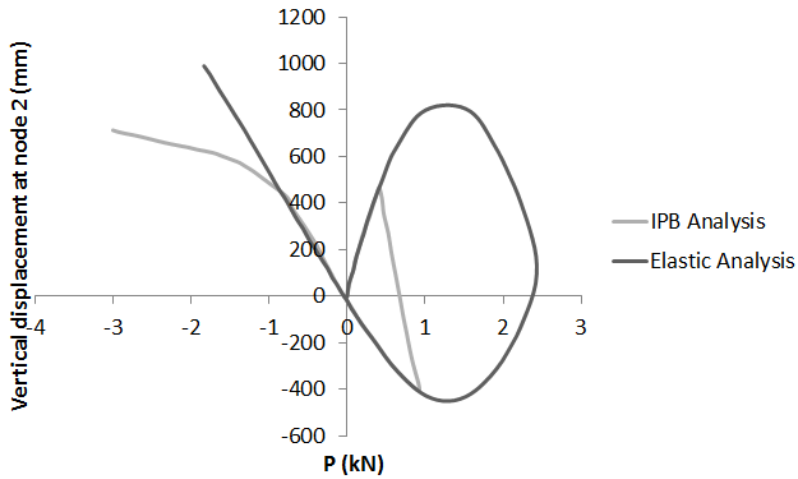

Figure 7. Load-displacement curves of circular dome truss at node 2

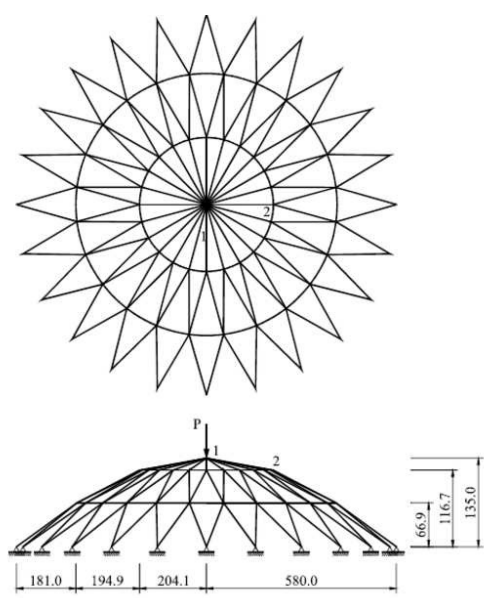

Figure 6. Circular dome truss, dimensions are given in cm.

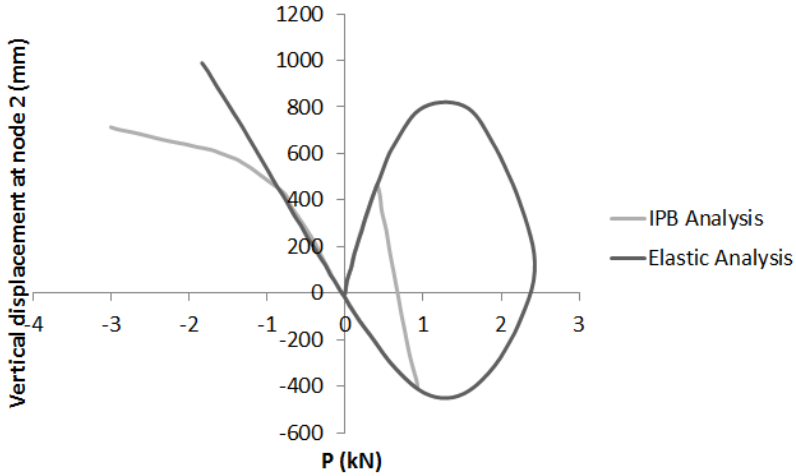

Figure 7. Load-displacement curves of circular dome truss at node 2

Table 2: Comparison of CPU Time (sec) for Example 2

		Analysis	
method	preconditioner	Elastic	IPB
N-R		26.8613	87.2952
N-R+BCG		10.3348	36.1059
	diag	9.5504	32.8564
NR+PBCG	p1	8.3961	29.6068
	p2	7.7956	27.4405
IBCG		7.1967	27.4122
	diag	6.6814	25.1918
PIBCG	p1	6.9661	22.9714
	p2	5.5567	21.4912

Example 3:

Schewdeler's truss, shown in Figure 8 having 264 members and 97 nodes with pin supports at the outer nodes, is taken from the Greco et al. [16]. The axial stiffness for all members is $EA = 640 \times 10^3 kN$ and their moment of inertia is $I = 1cm^4$ and $F_y = 25kN/cm^2$.

The truss is subjected to a single central concentrated load. $P = 50kN$

For Eq. (14) parameters $\Delta\lambda_1^1 = 0.01$, $\lambda_{\max} = 1$, $\gamma = 0.1$, $J_D = 2$, $J_{\max} = 100$ are selected.

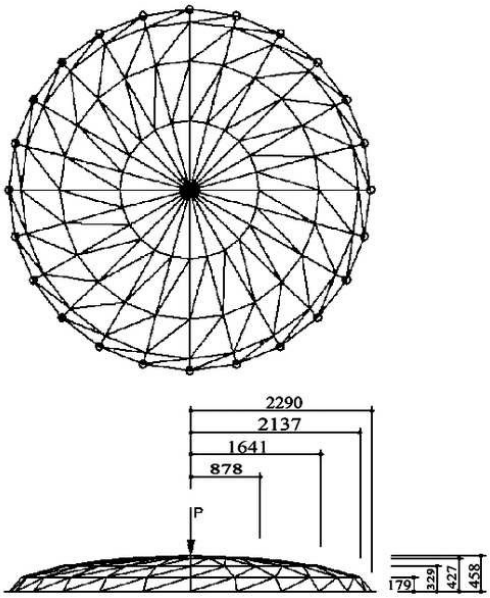

Figure 8. Schewdeler's dome truss, dimensions are given in cm.

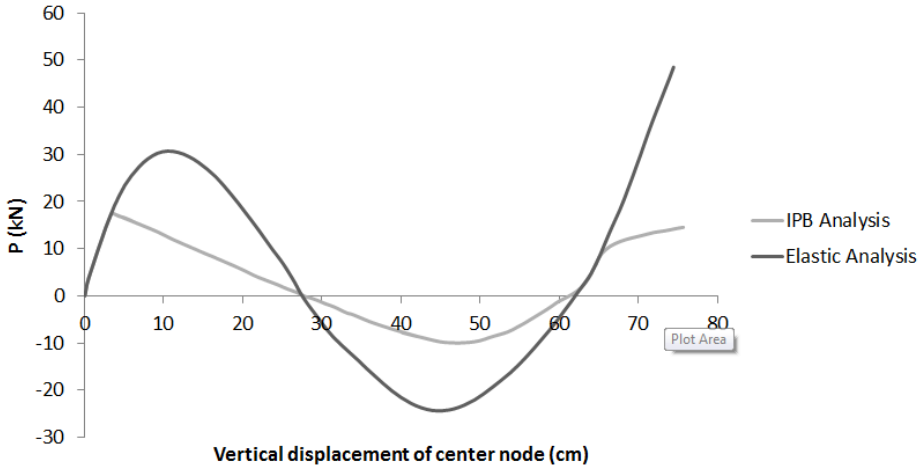

Figure 9. Central node vertical load–displacement

Table 3 shows a summary of the CPU time taken for application of classic Newton–Raphson method and four other algorithms discussed earlier, in 2 cases of analysis.

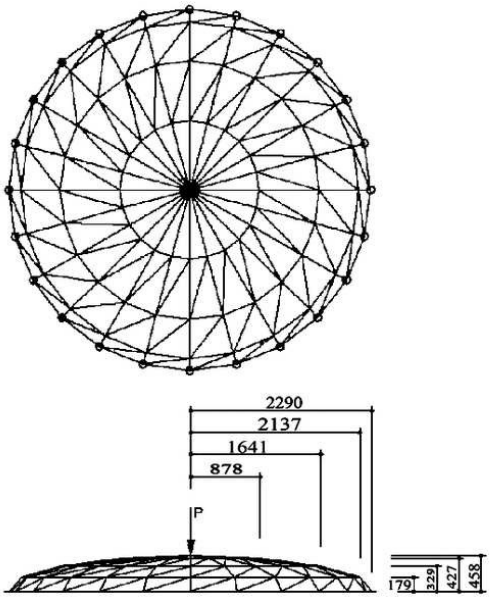

Figure 8. Schewdeler's dome truss, dimensions are given in cm.

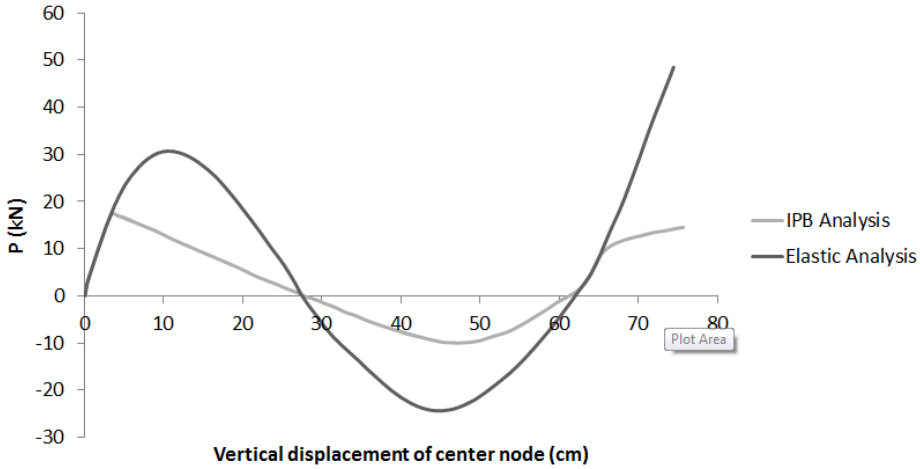

Figure 9. Central node vertical load–displacement

Table 3 shows that minimum computing time are related to the PIBCG algorithm.

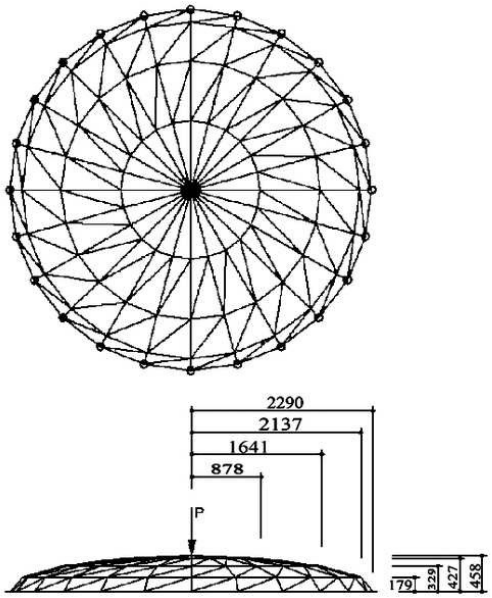

Figure 8. Schewdeler's dome truss, dimensions are given in cm.

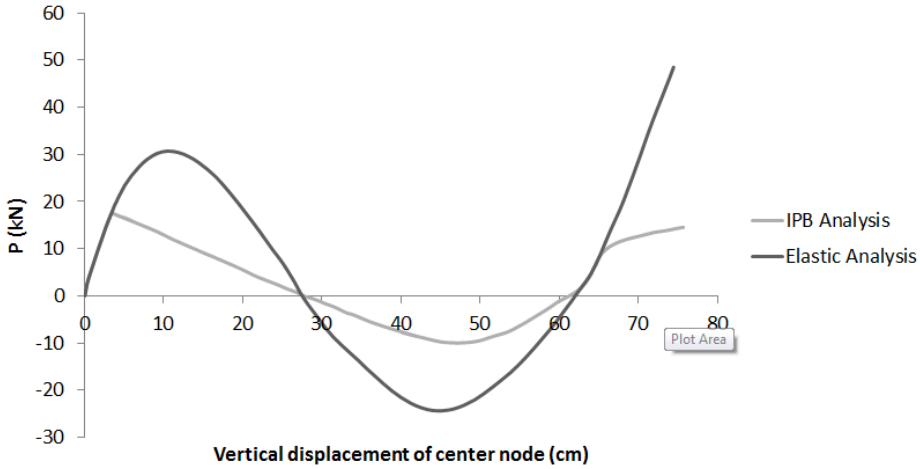

Figure 9. Central node vertical load–displacement

Table 3: Comparison of CPU Time (sec) for Example 3

		Analysis	
method	preconditioner	Elastic	IPB
N-R		115.6921	184.1164
N-R+BCG		41.2246	55.9401
	diag	39.1434	52.9055
NR+PBCG	p1	33.0622	45.8709
	p2	30.3414	42.5145
IBCG		35.7025	42.1109

	diag	32.6499	40.5104
PIBCG	p1	29.5974	34.9087
	p2	27.5623	32.5096

5. CONCLUSIONS

Most of the conventional techniques for analyzing nonlinear structures are inefficient and time-consuming. A new method was presented to analyze nonlinear structures in order to achieve lower computing time without losing accuracy. The proposed method analyzes structures with complex behaviors, including unloading, snap-through buckling, and inelastic post-buckling analyses. Applying iterative methods reduces the computing time significantly while the accuracy of the solution is at a desirable level. In this work, preconditioning techniques are applied to the BCG and IBCG method. It was shown that applying a suitable preconditioner in the PBGC and PIBCG method results in the best performance among all the five studied methods. A computer program was developed based on the algorithm discussed in this paper. Developed computer program was used to analyze different numerical examples. In fact, a lot of numerical tests were performed to show the ability of the proposed algorithms. Results indicate that applying the suggested preconditioner provides a useful practical means for nonlinear analysis of large structural problems.

REFERENCES

1. Chan S. Non-linear behavior and design of steel structures, *Journal of Constructional Steel Research*, **57**(2001) 1217-31.
2. Chan SL, Chui PT. *Non-Linear Static and Cyclic Analysis of Steel Frames with Semi-Rigid Connections*, San Diego, CA, Elsevier, 2000.
3. Kwasniewski L. Complete equilibrium paths for Mises trusses, *International Journal of Non-Linear Mechanics*, **44**(2009) 19-26.
4. Riks E. An incremental approach to the solution of snapping and buckling problems, *International Journal of Solids and Structures*, **15**(1979) 529-51.
5. Crisfield MA. A fast incremental/iterative solution procedure that handles snap-through, *Computers and Structures*, **13**(1981) 55-62.
6. Bellini PX. The concept of snap-buckling illustrated by a simple model, *International Journal of Non-Linear Mechanics*, **7**(1972) 643-50.
7. Ragon S, Gürdal Z, Watson LT. A comparison of three algorithms for tracing nonlinear equilibrium paths of structural systems, *International Journal of Solids and Structures*, **39**(2002) 689-98.
8. Eriksson A. Structural instability analyses based on generalised path-following, *Computer Methods in Applied Mechanics and Engineering*, **156**(1998) 45-74.
9. Saffari H, Fadaee M, Tabatabaei R. Nonlinear analysis of space trusses using modified normal flow algorithm, *Journal of Structural Engineering, ASCE*, **134**(2008) 998-1005.

10. Tabatabaei R, Saffari H. Large strain analysis of two dimensional frames by the normal flow algorithm, *Structural Engineering and Mechanics*, **36**(2010) 529-44.
11. Tabatabaei R, SaffariH, Fadaee M. Application of normal flow algorithm in modal adaptive Pushover analysis, *Journal of Constructional Steel Research*, **65**(2009) 89-96.
12. Petković MS, Petković LD. Families of optimal multipoint methods for solving nonlinear equations: a survey, *Applicable Analysis and Discrete Mathematics*, **4**(2010) 1-22.
13. Chun C. A new iterative method for solving nonlinear equations, *Applied Mathematics and Computation*, **178**(2006) 415-22.
14. Babajee D, Dauhoo M. An analysis of the properties of the variants of Newton's method with third order convergence, *Applied Mathematics and Computation*, **183**(2006) 659-84.
15. Papadrakakis M, Gantes C. Truncated Newton methods for nonlinear finite element analysis, *Computers and Structures*, **30**(1988) 705-14.
16. Greco M, Gesualdo F, Venturini W, Coda H. Nonlinear positional formulation for space truss analysis, *Finite Elements in Analysis and Design*, **42**(2006) 1079-86.
17. Thai HT, Kim SE. Large deflection inelastic analysis of space trusses using generalized displacement control method, *Journal of Constructional Steel Research*, **65**(2009) 1987-94.
18. Saffari H, Mansouri I. Non-linear analysis of structures using two-point method, *International Journal of Non-Linear Mechanics*, **46**(2011) 834-40.
19. Saffari H, Mirzai NM, Mansouri I. An accelerated incremental algorithm to trace the nonlinear equilibrium path of structures, *Latin American Journal of Solids and Structures*, **9**(2012) 425-42.
20. Cadou JM, Duigou L, Damil N, Potier-Ferry M. Convergence acceleration of iterative algorithms Applications to thin shell analysis and Navier–Stokes equations, *Computational Mechanics*, **43**(2009) 253-64.
21. Mansouri I, Saffari H, A fast hybrid algorithm for nonlinear analysis of structures, *Asian Journal of Civil Engineering*, **15**(2014) 213-229.
22. Saffari H, Baghlani A, Mirzai NM, Mansouri I. A New approach for convergence acceleration of iterative methods in structural analysis, *Journal of Computational Methods*, ASCE, **10**(2013) 16p.
23. Shewchuk JR. *An introduction to the conjugate gradient method without the agonizing pain*, Carnegie Mellon University, Pittsburgh, PA, City, 1994.
24. Van Loan GGC, Golub GH. *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 1996.
25. Fu W, Ho S, Li H, Wong HCC. An effective method to reduce the computing time of nonlinear time-stepping finite-element magnetic field computation, *IEEE Transactions on Magnetics*, **38**(2002) 441-44.
26. Yang J. Newton-conjugate-gradient methods for solitary wave computations, *Journal of Computational Physics*, **228**(2009) 7007-24.
27. Saffari H, Mirzai NM, Mansouri I, Bagheripour MH. Efficient Numerical Method in Second-Order Inelastic Analysis of Space Trusses, *Journal of Computing in Civil Engineering*, *ASCE* **27**(2013) 129-38.
28. Saad Y. *Iterative Methods for Sparse Linear Systems*, SIAM, 2003.

29. Van der Vorst HA. *Iterative Krylov methods for large linear systems*, Cambridge University Press, 2003.
30. Kassimali A, Bidhendi E. Stability of trusses under dynamic loads, *Computers and Structures*, **29**(1988) 381-92.
31. Hill CD, Blandford GE, Wang ST. Post-buckling analysis of steel space trusses, *Journal of Structural Engineering*, *ASCE* **115**(1989) 900-19.
32. Benzi M. Preconditioning techniques for large linear systems: a survey, *Journal of Computational Physics*, **182**(2002) 418-77.
33. Ramesh G, Krishnamoorthy C. Inelastic post-buckling analysis of truss structures by dynamic relaxation method, *International Journal for Numerical Methods in Engineering*, **37**(1994) 3633-57.