University Publication Centre (UPENA)
UNIVERSITI TEKNOLOGI MARA

ESTEEM

Academic Journal UiTM Pulau Pinang

Volume 5, Number 1 June 2009 ISSN 1675-7939

Esteem Academic Journal is jointly published by the Universiti Teknologi MARA, Pulau Pinang and University Publication Centre (UPENA), Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia.

The views, opinions and technical recommendations expressed by the contributors and authors are entirely their own and do not necessarily reflect the views of the editors, the Faculty or the University.

ESTEEM

Academic Journal UiTM Pulau Pinang

Volume 5, Number 1 **June 2009** **ISSN 1675-7939**

Foreword

Alhamdulillah. First of all a big thank you and congratulations to the Editorial Board of *Esteem Academic Journal* of Universiti Teknologi MARA (UiTM), Pulau Pinang for their diligent work in producing this issue. I also would like to thank the academicians for their contributions and the reviewers for their meticulous vetting of the manuscripts. A special thanks to University Publication Centre (UPENA) of UiTM for giving us this precious opportunity to publish this first issue of volume 5. In this engineering issue we have upgraded the standard of the manuscript reviewing process by inviting more reviewers from our university as well as other universities in Malaysia. We have embarked from previous volume to establish a firm benchmark and create a journal of quality and this current issue remarks a new height of the journal quality. Instead of publishing once in every two years, now *Esteem* publishes two issues annually.

In this issue, we have compiled an array of 13 interesting engineering research and technical based articles for your reading. The first article is entitled "The Response of Tube Splitting on Circular Tubes by Using Various Types of Semi-angles Dies and Slits". The authors, Mohd Rozaiman Aziz and Roslan Ahmad investigated the axial splitting and curling behavior of aluminum circular metal tubes which was compressed axially under static loading using three types of dies with different semi-angles. The authors concluded that the introduction of slit to the specimen is necessary to initiate slitting rather than inversion.

Salina Budin, Aznifa Mahyam Zaharudin, and Sugeng Priyanto presents a model of energy conversion and impact energy generation during collision based on free falling experiment, which is closely resembles direct collision between ball and inner wall of the vial. Simulation results from the proposed impact energy model demonstrated that the impact energy generated during the collision is strongly influenced by the thickness of the work materials and reaches zero at certain value of the work materials thickness, which increases with an increase of falling height.

Salina Alias, Caroline Marajan and Mohamad Azrul Jemain wrote an article that looks at adsorption of zinc from waste water using bladderwort (*Utricularia vulgaris*). In batch adsorption studies, data show that dried bladderwort has considerable potential in the removal of metal ions from aqueous solution. The fourth article written by

Muhammad Khusairi Osman et al. looked at 3D object recognition using affine moment invariants and Multiple Adaptive Network Based Fuzzy Inference System (MANFIS). The experimental results show that Affine Moment Invariants combined with MANFIS network attain the best performance in both recognitions, polyhedral and free-form objects.

The article entitled "Construction Waste Management Methods Used by Contractors in the Northern Region" authored by Siti Hafizan Hassan, Nadira Ahzahar and Mohd Nasrul Nizam Nasri reports an ongoing study on the use of construction waste management methods by contractors and its impact on waste reduction in the Northern Region. In conclusion, the sizing and amount of materials to be ordered to reduce wastage is significant in reducing construction waste generation waste, alleviating the burden associated with its management and disposal. The sixth article by Muhammad Sofian Abdullah et al. examined on the performance of Performance of Palm Oil Fuel Ash (POFA) with lime as stabilizing agent for soil improvement. The authors concluded that POFA can be used to treat the silty soil as well as to reduce the environmental problem.

The seventh article penned by Soffian Noor Mat Saliah, Noorsuhada Md. Nor and Megat Azmi Megat Johari presents the results of an experimental study on the interfacial bond strength (IBS) of polypropylene fiber concrete (PFC). It was found that the interfacial bond strength between concrete and reinforcement bar was not affected by the inclusion of polypropylene fibers. However, concrete containing fibers exhibited no breaking of concrete and no debonding of reinforcement. The article by Juliana Zaabar and Rusnani measures, evaluates and analyzes the network link performance of fiber optic cable using OTDR. The authors suggested that the major loss for these measurements is connector loss. Preventive maintenance will increase the life time of fiber optic. From some of the findings, the PVC dust cap has been identified as a main source of contamination for the SC connector.

The article entitled "Symbolic Programming of Finite Element Equation Solving for Plane Truss Problem" by Syahrul Fithry Senin proposed a plane truss problem to be solved by finite element method using MAPLE 12 software. The numerical solution computed by the author was almost matched with the commercial finite element software solution, LUSAS. The tenth article by Nor Azlan Othman, Nor Salwa Damanhuri and Visakan Kadirkamanathan presents a detail review of fault diagnosis in rotating machinery using pattern recognition technique. The authors proposed a solution based on artificial neural network (ANNs) which is Multi-Layer Perceptron (MLP). The authors concluded that the proposed methods are suitable for rotating machinery on fault detection and diagnosis.

The eleventh article is entitled "RAS Index as a Tool to Predict Sinkhole Failures in Limestone Formation Areas in Malaysia". Damanhuri Jamalludin et al. found that, using the RAS classification method, the prediction of sinkhole occurrences can be easily be made by simply knowing the weekly rainfall especially in areas having limestone as the bedrock. The twelfth by Muhammad Hafeez Osman et al. explores cases regarding the histories of rock slope repair and stabilization of unstable boulder along the road from Bukit Cincin to Genting Highland and along the road from Gap to Fraser Hill. The last article is "Soil Nail and Guniting Works in Pahang". The authors, Damanhuri Jamalludin et al. concluded that if the stability of the embankment needs to be improved, soil nails can be installed and embankment surface can be covered with gunite to prevent erosion.

We do hope that you not only have an enjoyable time reading the articles but would also find them useful. Thank you.

Mohd Aminudin Murad
Chief Editor
Esteem, Vol. 5, No. 1, 2009
(Engineering)

Symbolic Programming of Finite Element Equation Solving for Plane Truss Problem

Syahrul Fithry Senin
Faculty of Civil Engineering
Universiti Teknologi MARA (UiTM), Malaysia
Email: syahrul573@ppinang.uitm.edu.my

ABSTRACT

Due to the significant progress in computer programming technology and finite element method that have been achieved in the recent years, the use of symbolic programming techniques in engineering computational process gain more and more importance. Symbolic programming technology in scientific programming offer easier, faster and compact routine procedure writing as compared the available high-level programming language, such as FORTRAN, which was previously one of the traditional computer languages to program the finite element procedures.

In this paper, a plane truss problem was proposed and solved by finite element via MAPLE 12 software, a symbolic programming language. The whole digital process allows the computer user to appreciate and experience the real-time finite element solution process of the plane truss problem. The plane truss was discretized by the computer user and analyzed by special commands that imitating the human natural instructions. The local and global stiffness matrix, the assembled global stiffness matrix of the plane truss problem can generate and visualized after suitable human-like command entered and executed by the computer. The incorporation of boundary condition on the assembled stiffness matrix needs the user to specify the constrained global degrees of freedom of the problem on the assembled global stiffness matrix. Finally, after imposing the relevant loads on the plane truss nodes, the problem was solved by MAPLE solver to obtain the nodal displacements and element stresses. Verification of those results with the present numerical results and

ISSN 1675-7939

LUSAS software was conducted and good agreement was found up to maximum error of 6.2 × 10^{-3} percent.

Keywords: *Plane Truss, Symbolic Programming, Finite Element, LUSAS, Maple*

Introduction

Finite element analysis is a powerful numerical method for solving complex problems of engineering and mathematical physics since its first development in the year of 1941. The classical principles lies on finite element analysis are still similar despite the new and emerging structural applications. Probably, one of the new recent structural applications was a truss structure with random parameters that been successfully analyzed by finite element analysis (Huang, Su); still using the mandatory classical finite element principles. Zhou and Li (2005) studied the stress and strain at the nodes of the Mitchell truss in three-dimensional using finite element method and found the results agreed after several iterations. Nevertheless, none of them mentioned explicitly to use symbolic programming language to solve the problems.

The use of computer algebra system (CAS) or simply known as symbolic programming system has been a matter of interest for finite element development. Most probably the first related works that using CAS was the work of Wilson and Clough in the year of 1963; a system called as SMIS (Symbolic Matrix Interpretive System) was developed for the purpose of teaching the static and dynamic analysis of structures. A more advanced finite element research problems such as incompressible flow problem, crystal growth during solidification and computation of fluid and heat flow in a weld pool have been formulated symbolically using symbolic computation software by Gustav, Robert and Christian (1999).

The key idea to use CAS in the finite element teaching and research in the works of Wilson, Clough and Gustav (1999) is the need to enable active learning and hands-on experimentation on problems among the students or researchers via its interactivity features. Therefore, in this paper, the main critical of finite element procedures such as the local stiffness matrix derivation, the assembly of global stiffness matrix, the incorporation of boundary condition and solving the simultaneous linear equations of a plane truss problem have been coded in Maple 12 and its interactivity features will be addressed in the following sections in this paper.

The Structure and Explanation on the Programming Works

A general FEA computer program, written in MAPLE 12 language, was developed to solve a plane truss problem. In this programming work, four main Maple procedures were developed specifically to execute FEA procedures on solving truss problems. The four main procedures are listed as follows:

- LengthofTruss
- StiffnessofTruss
- GlobalAssemble
- GlobalTrussForce

Those four procedures are able to express the solution of interest, i.e. the displacement of the truss node, and the derived variable, i.e. truss element forces, either in symbolic or in numerical forms. However, in this paper, numerical form of solution of the problem is only discussed. Table 1 shows the list of the developed procedures, the general description, the argument variables and the output of each procedure.

The sequence of analysis is initiated by the execution of the first procedure and followed by the next procedure and ended by the fourth procedure.

Table 1: The Brief Description of the Main Procedures

No	Procedure's name	Argument variables	Output
1	LengthofTruss	(x1,y1) – First node coordinate (x2,y2) – Second node coordinate	Element Length
2	StiffnessofTruss	E – Young's Modulus A – Cross sectional area L – Element length Thetha – Element angle	Element Stiffness Matrix
3	GlobalAssemble	K – Global stiffness k – Local assembled matrix i – The first node of element j – The second node of element	Assembled Global Stiffness Matrix
4	GlobalTrussForce	E – Young's Modulus A – Cross sectional area L – Element Length Thetha – Element angle u – Local element displacement	Local Element Forces

The incorporation of boundary condition to the assembled global stiffness matrix was conducted after the execution of the fourth procedure. The matrix size reduction process from the assembled global stiffness matrix into reduced assembled global stiffness, $[K]_{reduced}$ matrix was symbolically done using the built-in MAPLE 12 command known as *DeleteRow or DeleteColumn.*

DeleteRow(A,L,outopts) or DeleteColumn(A,L,outopts) is a built-in command in the linear algebra Maple 12 library to return a submatrix K obtained by deleting the rows or columns determined by parameter L. As an example, consider a global stiffness matrix K of the plane truss problem is defined by

$$[K]=\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Let say, the constrained node of the global stiffness matrix K is located at the third row and the third column of the matrix. $[K]_{reduced}$ is the reduced form of the assembled global stiffness matrix after the removal of corresponding rows and column associated with the constrained nodes of the stiffness matrix. In order to remove the constrained node of the matrix, *DeleteRow(K,3)* and *DeleteColumn(K,3)* are executed on the matrix and the reduced form of the assembled global stiffness matrix, $[K]_{reduced}$ is

$$[K]_{reduced}=\begin{bmatrix} 1 & 2 \\ 4 & 5 \end{bmatrix}$$

Similarly, the reduced global force vector, {f}, also has to be defined in order to represent the magnitude of external forces that acting on unconstrained nodes. The unconstrained degrees of freedom vector, $\{u\}$, is related to the reduced global force vector, {f}, by the following equation;

$$\{f\}=[K]_{reduced}\{u\}$$

The global displacement matrix $\{u\}$ of the plane truss problem is to be solved by MAPLE 12 using the built-in command LinearSolve. The corresponding element forces will be computed by multiplying the global stiffness matrix under the consideration with the global degrees of freedom matrix.

Results and Discussion

A plane truss structure with three elements as shown in Figure 1 is under the application of a vertical point load of 1000 N at node 2. The truss material is an isotropic type with Young's Modulus value as 70 × 109 N/m^2. Each of the truss members has a cross sectional area of 0.1 m^2. Node number 3 is constrained to move along horizontal axis and node 1 is pinned supported. The analysis of local and global displacements and its element forces was executed by the symbolic programming MAPLE 12.

Before being analysed by the procedures, the user has to discretise each element to three nodes and three local elements. In Figure 1, the circled number denotes the node number and the number inside the box denotes the element number. The global coordinate system x-y and x'-y' are defined to show the orientation of the global and local member of the truss. In order to define each element connectivity and orientations, Table 2 shows the nodal connectivity of each element.

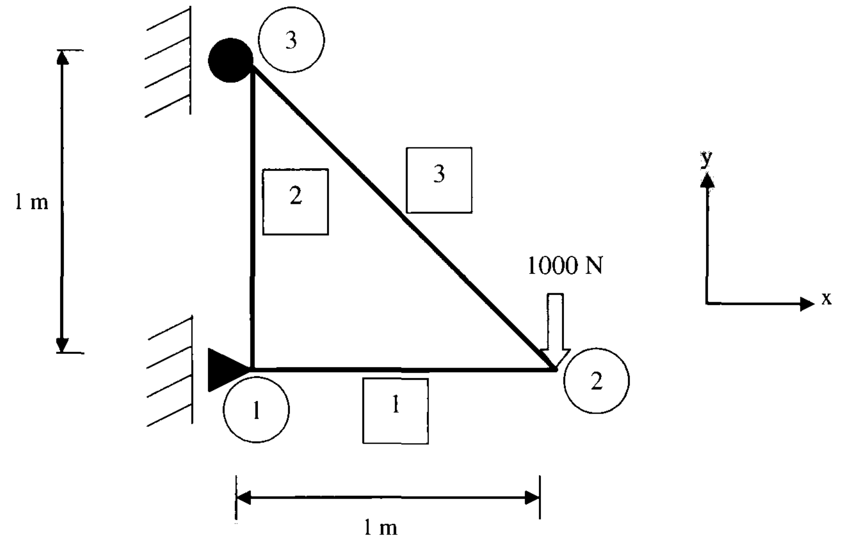

Figure 1: A Three Member Truss Structure

Table 2: Nodal Connectivity

Element no	Starting node	Ending node
1	1	2
2	1	3
3	2	3

Using the first procedure, *LengthofTruss*, the length of each element is computed by submitting the Young's Modulus, E, the cross sectional areas, A, the orientation angle, thetha. The command and the output of the each length, Li,, where i is the element number are shown below;

```
L1:=LengthofTruss(0,0,1,0);
```

$$L1 := 1$$

```
L2:=LengthofTruss(0,0,0,1);
```

$$L2 := 1$$

```
L3:=LengthofTruss(1,0,0,1);
```

$$L3 := \sqrt{2}$$

Basically, *LengthofTruss* procedure will compute the length of each element length using the following equation

$$\text{LengthofTruss} = \sqrt{(x_2 - x_1)^2 + (y_2 + y_1)^2}$$

The 4 × 4 global stiffness matrix of each element is generated by typing a procedure name called as *StiffnessofTruss*. The *StiffnessofTruss* procedure will compute the global stiffness matrix, [K], based on the following finite element equations;

$$[K] = [T]^T [k'][T]$$

where

$$[k'] = \frac{EA}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

$$[T] = \begin{bmatrix} [CS] & [ZERO] \\ [ZERO] & [CS] \end{bmatrix}$$

$$[CS] = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

$$[ZERO] = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

In the above statement, E, A, L and θ is the Young's Modulus, the cross sectional areas, the element's length and the orientation of each member. The output of the first element after executing *StiffnessofTruss* procedure is as follows;

```
k1:=map(evalf,StiffnessofTruss(E,A,L1,theta1));
```

$$kl := \begin{bmatrix} 0.7000000\ 10^7 & 0. & -0.7000000\ 10^7 & 0. \\ 0. & 0. & 0. & 0. \\ -0.7000000\ 10^7 & 0. & 0.7000000\ 10^7 & 0. \\ 0. & 0. & 0. & 0. \end{bmatrix}$$

The next crucial step in the finite element formulation is the assembly of each global stiffness matrix with the matrix size of 2 times node numbers by 2 times node numbers. Each of the global stiffness matrix will be superimposed by adding up the corresponding rows and columns position simply on typing *GlobalAssemble* command as shown below;

```
GlobalAssemble(K,k1,1,2):
GlobalAssemble(K,k2,1,3):
GlobalAssemble(K,k3,2,3):
```

The computed array format of the assembled global stiffness matrix should be converted to matrix format to enable the user to visualize the numerical solution of the assembled stiffness matrix.

```
K:=convert(K,Matrix);
```

$[0.7000000000\ 10^{10}, 0., -0.7000000000\ 10^{10}, 0., 0., 0.]$

$[0., 0.7000000000\ 10^{10}, 0., 0., 0., -0.7000000000\ 10^{10}]$

$[-0.7000000000\ 10^{10}, 0., 0.9474873734\ 10^{10}, -0.2474873734\ 10^{10},$
$-0.2474873734\ 10^{10}, 0.2474873734\ 10^{10}]$

$[0., 0., -0.2474873734\ 10^{10}, 0.2474873734\ 10^{10}, 0.2474873734\ 10^{10},$
$-0.2474873734\ 10^{10}]$

$[0., 0., -0.2474873734\ 10^{10}, 0.2474873734\ 10^{10}, 0.2474873734\ 10^{10},$
$-0.2474873734\ 10^{10}]$

[0. , -0.7000000000 10^{10} , 0.2474873734 10^{10} , -0.2474873734 10^{10} , -0.2474873734 10^{10} , 0.9474873734 10^{10}]

The constrained degrees of freedom of the plane truss problem are associated with the first, second and the third rows and columns of the assembled global stiffness matrix. To remove the whole constrained degrees of freedom, each of the above-mentioned rows and columns are deleted using the DELETEROW and DELETECOLUMN command. The principle lies on using these commands have been briefly explained in this paper. The process of deleting each rows and columns is shown below and finally the system stiffness matrix, *Reduced_K*, of the whole plane truss is generated with 3x3 matrix size.

```
Reduced_K:=DeleteRow(K,[1..2]);
```

[-0.700000000000000000 10^{10} , 0. , 0.947487373400000000 10^{10} , -0.247487373400000000 10^{10} , -0.247487373400000000 10^{10} , 0.247487373400000000 10^{10}]
[0. , 0. , -0.247487373400000000 10^{10} , 0.247487373400000000 10^{10} , 0.247487373400000000 10^{10} , -0.247487373400000000 10^{10}]
[0. , 0. , -0.247487373400000000 10^{10} , 0.247487373400000000 10^{10} , 0.247487373400000000 10^{10} , -0.247487373400000000 10^{10}]
[0. , -0.700000000000000000 10^{10} , 0.247487373400000000 10^{10} , -0.247487373400000000 10^{10} , -0.247487373400000000 10^{10} , 0.947487373400000000 10^{10}]

```
Reduced_K:=DeleteColumn(Reduced_K,[1..2]);
```

[0.947480000000000000 10^{10} , -0.247480000000000000 10^{10} , -0.247480000000000000 10^{10} , 0.247480000000000000 10^{10}]
[-0.247480000000000000 10^{10} , 0.247480000000000000 10^{10} , 0.247480000000000000 10^{10} , -0.247480000000000000 10^{10}]

[-0.247480000000000000 10^{10} , 0.247480000000000000 10^{10} , 0.247480000000000000 10^{10} , -0.247480000000000000 10^{10}]
[0.247480000000000000 10^{10} , -0.247480000000000000 10^{10} , -0.247480000000000000 10^{10} , 0.947480000000000000 10^{10}]

```
Reduced_K:=DeleteColumn(Reduced_K,[3..3]);
```

$[0.947480000000000000 \ 10^{10}, -0.247480000000000000 \ 10^{10}, 0.247480000000000000 \ 10^{10}]$
$[-0.247480000000000000 \ 10^{10}, 0.247480000000000000 \ 10^{10}, -0.247480000000000000 \ 10^{10}]$

$[-0.247480000000000000 \ 10^{10}, 0.247480000000000000 \ 10^{10}, -0.247480000000000000 \ 10^{10}]$
$[0.247480000000000000 \ 10^{10}, -0.247480000000000000 \ 10^{10}, 0.947480000000000000 \ 10^{10}]$

```
Reduced_K:=DeleteRow(Reduced_K,[3..3]);
```

$[0.947480000000000000 \ 10^{10}, -0.247480000000000000 \ 10^{10}, 0.247480000000000000 \ 10^{10}]$
$[-0.247480000000000000 \ 10^{10}, 0.247480000000000000 \ 10^{10}, -0.247480000000000000 \ 10^{10}]$

$[0.247480000000000000 \ 10^{10}, -0.247480000000000000 \ 10^{10}, 0.947480000000000000 \ 10^{10}]$

The load vector, f, of the problem is defined by converting the vertical point load of 1000 N acting downwards in a form shown as below;

```
f:=Matrix( [ [0],[-1000],[0] ]);
```

$$f := \begin{bmatrix} 0 \\ -1000 \\ 0 \end{bmatrix}$$

In order to solve the plane truss problem to find its global displacement vector, u, the following equation should be solving using matrix algebra knowledge;

$$\{f\} = [Reduced\_K]\{u\}$$

Maple 12 handles the above equation solving using *LinearSolve* command with the format shown below and the global displacement vector, u, in the unit of meter is obtained;

```
u:=LinearSolve(Reduced_K,f);
```

$$\begin{bmatrix} -0.142857142857142847 \;\; 10^{-6} \\ -0.689787342122884342 \;\; 10^{-6} \\ -0.142857142857142847 \;\; 10^{-6} \end{bmatrix}$$

Global element forces vector, F, can be obtained by multiplying the global stiffness matrix, K, with the global displacement vector, u, in the following sequence order as depicted below;

```
zero:=<<0>>;
```

$$zero := [\ 0]$$

```
U:=<zero|zero|u[1]|u[2]|zero|u[3]>;
```

$$[0\ ,0\ ,-0.142857142857142847 \;\; 10^{-6}\ ,-0.689787342122884342 \;\; 10^{-6}\ ,0\ ,\ -0.142857142857142847 \;\; 10^{-6}\ ]$$

```
F:=MatrixMatrixMultiply(K,U);
```

$$\begin{bmatrix} 999.99999999999988 \\ 999.99999999999988 \\ 0.510702591327572009 \;\; 10^{-14} \\ -999.99999999999988 \\ -999.99999999999988 \\ 0.510702591327572009 \;\; 10^{-14} \end{bmatrix}$$

The stresses of each element, sigma1 and sigma2 (in N/m^2), able also to be computed simply by executing *TrussStress* procedure with its corresponding argument variables that can be detected from the following commands;

```
sigma1:=TrussStress(E,L1,theta1,u1);
```

$$[-10000.]$$

```
sigma2:=evalf(TrussStress(E,L2,theta2,u2));
```

$$[-10000.]$$

```
sigma3:=evalf(TrussStress(E,L3,theta3,u3));
```

$$[14143.]$$

The accuracy of numerical values of the displacement vector, u, and the each element stresses, sigma1, sigma2 and sigma3, were compared with LUSAS software and the results were summarized in Table 3.

Table 3: Comparison the Present Numerical Values with LUSAS

Parameters	Present numerical values	LUSAS	Error (%)
Node 2 horizontal displacement	-1.42857×10^{-6} m	-1.42857×10^{-6} m	0
Node 2 vertical displacement	-6.8978×10^{-6} m	-6.8978×10^{-6} m	0
Node 3 vertical displacement	-1.42857×10^{-6} m	-1.42857×10^{-6} m	0
Element 1 stress	-10000.0000 N/m^2	-10000 N/m^2	0
Element 2 stress	-10000.0000 N/m^2	-10000 N/m^2	0
Element 3 stress	-14143 N/m^2	-14142.135 N/m^2	6.2×10^{-3}

From Table 3, it was shown that the present numerical solution of horizontal and vertical displacement at node 2 and 3 and element stresses are almost exactly matched with the solution given by LUSAS software. The maximum percentage of error is limited to 6.2×10^{-3} for the numerical results of the third element stress. However, the solution provided by the present work enables active learning and hands-on experimentation on the problems as compared to LUSAS software. This is explained by the need of the user to simulate the finite element solving mechanism in correct sequence by executing suitable designed procedure. Using LUSAS alone, the user does not have any chances to experience and learn the finite element solving process due to instant solution provided by the software. Therefore, the programmed symbolic work can be deemed to enhance the learning process of finite element specifically on the plane truss problem.

Conclusion

A symbolic computer program that analyzing plane truss structure has been coded using MAPLE 12. The global stiffness matrices, the assembled global stiffness matrix and the reduced assembled stiffness matrix of the computer program can be viewed as the "book-style display". The computational of the above-mentioned matrix can be accessed by the learners and instructors by typing the procedure's name with the correct required argument variables. The numerical solution computed by the work was almost exactly matched with the commercial finite element software solution, LUSAS. The maximum percentage of error is limited to 6.2×10^{-3} for the numerical results of the third element stress. It is proposed that the developed computer program is used by the university instructors as the medium to teach FEA to the learners in an effective way.

References

Clough, R. W., & Wilson, E. L. (1999). Early Finite Element Research at Berkeley. *US National Conference on Computational Mechanics*, Aug 4-6, 1999.

Gustav, A., Robert, T., & Christian, W. (1999). Finite Element Simulations using symbolic computing, Mathematics and Computers in Simulation. *Elsevier Science*, *49*, 257-274.

Huang, B., & Suo, J. (2007). Geometrical nonlinear analysis of truss structures with random parameters utilizing recursive stochastic finite element method. *Lixue Xuebao/Chinese Journal of Theoretical and Applied Mechanics*, *39*(6), 835-842.

Zhou, K. M., & .Li, J. F. (2005). Forming Michell truss in three-dimensions by finite element. *Applied Mathematics and Mechanics (English Edition)*, *26*(3), 381-388.