

DEVELOPMENT OF OPEN SOURCE FINITE ELEMENT SOLVER “LD-FEM” FOR MODELING AND SIMULATION OF RUBBER MATERIALS

Sugeng Waluyo

Department of Engineering, University of Jenderal Soedirman
Jl. Mayjen Sungkono Km. 5 Blater Purbalingga 53371
E-mail: sugeng.waluyo@unsoed.ac.id

ABSTRACT

“LD-FEM” is an open source computer program working on the basis of finite element method (FEM) which is aimed to model and simulate large deformation in rubber materials. The kinematics of large deformation on the basis of the Total Lagrange framework is applied to linear 4-nodes tetrahedral element and then solved with Newton-Raphson iterative scheme. Furthermore, to obtain the material tangent stiffness directly from strain energy density functions, the Gill-Murray theory of numerical second derivative is used in LD-FEM. Finally, by using the Mooney-Rivlin strain energy function, the performance of LD-FEM is addressed for uniaxial tensile, shear and torsion loading tests. The results confirm the capability of LD-FEM to capture nonlinear behavior of the large deformation either with analytical or numerical approach on the material stiffness derivation with error less than 2%.

Keywords: large deformation, LD-FEM, material stiffness, numerical derivative, tetrahedral element.

1. INTRODUCTION

Efforts to develop finite element solver in Indonesia are set back to several authors such as [1] with software called SANSPRO, [2] for triangular element, [3] formulating 2D plane element, [4] for triangular element, [5] developing shell element and [6] focusing on beam reinforced concrete element. Before 1989 many Indonesian researchers especially coming from their graduate study in USA and Europe were strongly indicated to develop finite element solvers but literatures regarding their existence to the best of author knowledges are none. While only SANSPRO is still known as the only commercially available finite element solution, the remaining is considered as in-house solvers.

LD-FEM is an open source finite element analysis (FEA) program in 3D large deformation kinematics for educational purposes without considering programming optimization. It is mainly developed as a supplementary solver for the aforementioned finite element solvers. Programming activity of LD-FEM has been started from the mid of 2009. Here, FORTRAN 90 with G95 compiler [7] is used to construct the FEA solver for solid continua without pre- and post-processing capability. Rubber and soft tissue material simulation are two examples of typical problems involving large deformations which can be solved using LD-FEM.

At the beginning of section, this article deals with the main structure of LD-FEM programming code and the Newton-Raphson iteration scheme with load and displacement control as the nonlinear solver. The solid element used here is tetrahedral element with 4 nodes and linear shape function. The reason behind using the tetrahedral element is to accommodate mesh result from auto meshing program which is commonly found in complex geometry discretization. Hopefully, LD-FEM will be more user-friendly for less experienced users in advance meshing technique.

In the next section, the material stiffness evaluation from strain energy function W is given as an independent subroutine in order to facilitate user owned function. Here, the material stiffness is evaluated either by using analytical approach or numerical approach. The analytical approach will give an exact evaluation of the material stiffness but requires cumbersome analytical derivation with respect to the right Cauchy deformation gradient tensor C . Contrary to this, the numerical approach is widely known for its error prone nature resulting from approximation method and computer floating point despite being easier in computation. Hence, the numerical approach is preferred only as a validation tool for the analytical approach. From the computational aspect, the application of numerical approach combined with iterative scheme such as Newton method deals with important issues regarding stability and convergence rate.

In the last part, the performance of LD-FEM will be evaluated for cube geometry meshed with six tetrahedral elements. The result expected here is benchmarking between the numerical and the analytical approach in evaluation of material stiffness. The applied loadings are uniaxial tensile, shear and torsion in the particular face with fixed constraint at the opposite face. The strain energy function used is the isotropic Mooney-Rivlin material model. Additional test for LD-FEM to more complex problem, i.e., a rubber slliper rope, is presented in this section as well.

2. PROGRAMMING STRUCTURE AND FINITE ELEMENT FORMULATION

Programming structure of LD-FEM consists of 3 main subroutines: PRE_SOLID_FSFEM(), NONLIN_SOLVING(), and POST_SOLID_FSFEM(). They are designed as independent as possible from the change

of solid element type, material model, boundary condition and linear solver. The remaining subroutines are dedicated to perform specific tasks such as matrix algebra operations and finite element formulation.

The PRE_SOLID_FSFEM () collects the input parameters for simulation e.g., nodal, and element properties, material properties, boundary condition, and number of load increment, using an input file called ‘PreGMesh_3D.msh’. This subroutine can be further modified to control the simulation process, e.g., type of rubber or soft tissue material models. As the visualization tool, LD-FEM uses open source graphical user interfaces (GUI) software called Gmsh [8].

The main subroutine in LD-FEM is called NONLIN_SOLVING () which is responsible for performing Newton-Raphson iteration scheme. It contains four supporting subroutines namely LOAD_SOLID_FSFEM (), TET4nodes (), CONSTRAINT_SOLID_FSFEM () and SOLVER (). While functions of each this subroutine are detailed in Table 1, calling procedure of these is depicted in Figure 1.

Table 1. The function of supporting subroutines in NONLIN_SOLVING ()

Subroutine	Task for each load incremental
LOAD_SOLID_FSFEM()	collecting information regarding load applied in particular node and element
TET4nodes()	constructing stiffness matrix of individual 4-node tetrahedral element.
CONSTRAINT_SOLID_FSFEM()	collecting information regarding constraint applied in particular node and element
SOLVER()	solving system of linear equation

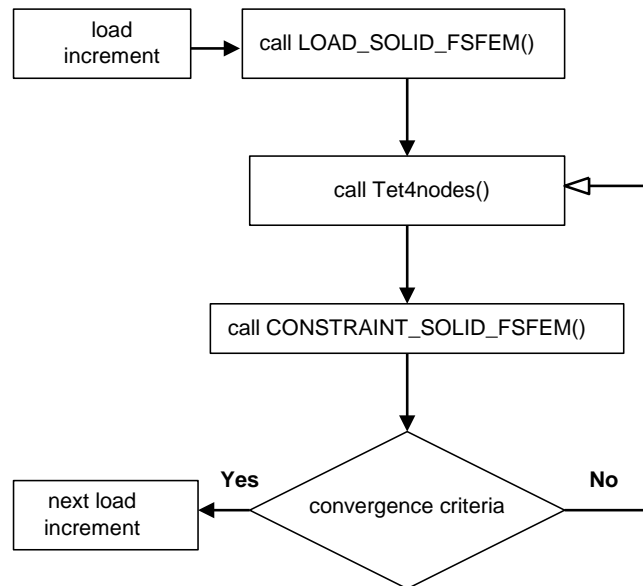


Figure 1. Subroutine calling procedure of Newton-Raphson scheme implementation in LD-FEM

The last subroutine namely POST_SOLID_FSFEM () provides a particular data for post-processing activities in the Gmsh. Here, it contains mainly node and element data from the pre-processing unit with additional data, i.e., deformation history with respect to load increment, coming up from the calculation process.

LD-FEM uses Newton-Raphson iteration scheme implemented in the NONLIN_SOLVING (). While the underlying finite element formulation will be explained briefly here, its detail derivation can be found in literatures such as [9] and [10]. The scheme is dedicated to solve the residual finite element vector \mathbf{r} over a body volume of V in the Total Lagrange framework \mathbf{X} as

$$\mathbf{r} = \int_V \frac{\partial \mathbf{v}^T}{\partial \mathbf{x}} \mathbf{F} \mathbf{T} dV - \mathbf{f}_{\text{ext}} \tag{1}$$

where \mathbf{v} , \mathbf{F} and \mathbf{T} are the virtual displacement, deformation gradient and the 2nd-Piola Kirchoff stress. The external force vector \mathbf{f}_{ext} is applied in the element node. By using load control method, the residual is derived with respect to each component of deformation gradient to obtain the incremental equation as

$$\mathbf{r}^n + \frac{\partial \mathbf{r}^n}{\partial (\mathbf{F})^n} \Delta(\mathbf{F}) = \mathbf{0} \quad (2)$$

with Δ as incremental vector in the global reference coordinate \mathbf{x} .
 Shortly, evaluating (2) yields

$$\left[\int_V \frac{\partial \mathbf{v}^T}{\partial \mathbf{x}} \mathbf{T}_M dV + \int_V \frac{\partial \mathbf{v}^T}{\partial \mathbf{x}} \mathbf{F}_M \mathbf{T}_M dV \right] \frac{\partial \Delta \mathbf{u}}{\partial \mathbf{x}} = - \int_V \frac{\partial \mathbf{v}^T}{\partial \mathbf{x}} \mathbf{F}_M \mathbf{T}_M dV + \mathbf{f}_{\text{ext}} \quad (3)$$

where \mathbf{T}_M and \mathbf{F}_M are two special matrices containing the component of 2ndPiola Kirchoff stress \mathbf{T} and deformation gradient \mathbf{F} , respectively. Note that the material stiffness can be found in \mathbf{T}_M by assuming similar relationship between modulus of elasticity and strain in the small deformation theory.

Formula in (3) is applied for linear 4-nodes tetrahedral element as depicted in Figure 2. Using parametric coordinate ξ, η, ϕ , the linear shape functions \mathbf{N} can be represented as $N_1 = \xi$, $N_2 = \eta$, $N_3 = \phi$, and $N_4 = 1 - \phi - \eta - \xi$ with four constraints i.e., $0 \leq \xi \leq 1$, $0 \leq \eta \leq 1$, $0 \leq \phi \leq 1$, and $N_1 + N_2 + N_3 + N_4 = 1$.

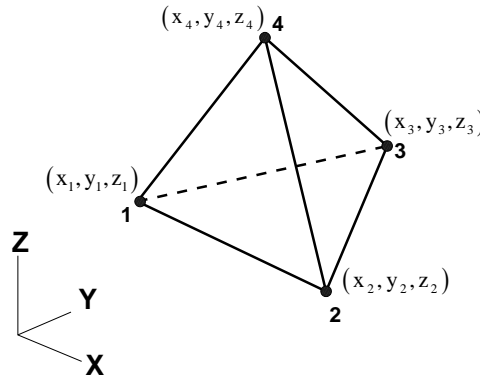


Figure 2. The 4-node linear tetrahedral element for LD-FEM in the global coordinate $\mathbf{x}(X, Y, Z)$

Using isoparametric approach, deformation and incremental deformation in the element can be represented as a function of its nodal deformation \mathbf{u}_{node} and incremental nodal deformation $\Delta \mathbf{u}_{\text{node}}$ as

$$\mathbf{u} = \mathbf{N} \mathbf{u}_{\text{node}} \quad (4)$$

$$\Delta \mathbf{u} = \mathbf{N} \Delta \mathbf{u}_{\text{node}} \quad (5)$$

Substitution of (4) and (5) to (2) gives the required final incremental equation for the programming task as

$$\mathbf{K} \Delta \mathbf{u}_{\text{node}} = \mathbf{f} \quad (6)$$

with

$$\mathbf{K} = \int_V \frac{\partial \mathbf{N}^T}{\partial \mathbf{x}} \mathbf{T}_M \frac{\partial \mathbf{N}}{\partial \mathbf{x}} dV + \int_V \frac{\partial \mathbf{N}^T}{\partial \mathbf{x}} \mathbf{F}_M \mathbf{T}_M \frac{\partial \mathbf{N}}{\partial \mathbf{x}} dV \quad (7)$$

and

$$\mathbf{f} = - \int_V \frac{\partial \mathbf{N}^T}{\partial \mathbf{x}} \mathbf{F}_M \mathbf{T}_M dV + \mathbf{f}_{\text{ext}} \quad (8)$$

3. NUMERICAL EVALUATION OF MATERIAL STIFFNESS

In order to numerically evaluate the first and second derivatives of strain energy density function $W(\mathbf{C})$ the Gill-Murray method [11] is implemented here. The first derivatives of the strain energy function with respect to each component in the right Cauchy deformation gradient matrix \mathbf{C} is known as the true stress σ_{ij} , i.e.,

$$\sigma_{ij} = \frac{\partial W}{\partial C_{ij}} = \frac{W(C_{ij} + \varepsilon) - W(C_{ij})}{\varepsilon} \tag{9}$$

where ε is the scalar perturbation parameter. Note for this section, the double indices, i.e., ‘ij’, ‘kl’ and ‘mn’, refers to row and column wise direction in a square matrix. With (9) in hand, the second derivatives of $W(\mathbf{C})$ can be evaluated by deriving (9) again to C_{ij} as

$$\frac{\partial^2 W}{\partial C_{ij} \partial C_{kl}} = \frac{1}{\varepsilon} \left[\frac{\partial W(C_{kl} + \varepsilon)}{\partial C_{ij}} - \frac{\partial W(C_{kl})}{\partial C_{ij}} \right]. \tag{10}$$

Substitution of (9) to (10) results in the material tangent stiffness of respective material as

$$C_{mn}^t = \frac{1}{\varepsilon^2} \left[\begin{array}{l} W_{mn}^*(C_{ij} + \varepsilon, C_{kl} + \varepsilon) - W_{mn}^*(C_{ij}, C_{kl} + \varepsilon) - \\ W_{mn}^*(C_{ij} + \varepsilon, C_{kl}) + W_{mn}^*(C_{ij}, C_{kl}) \end{array} \right]. \tag{11}$$

The W_{mn}^* is introduced here as a matrix representing change of the strain energy function with respect to ε . Since C_{mn}^t is a matrix with dimension of 6 x 6, $W_{mn}^*(\mathbf{C}, \varepsilon)$ must have double dimension, i.e., 12 x 12. The associated component of $W_{mn}^*(\mathbf{C}, \varepsilon)$ related to C_{mn}^t can be seen in Table 2.

Table 2. The associated component of $W_{mn}^*(\mathbf{C}, \varepsilon)$ for each component of C_{mn}^t

“m” for row “n” for column	W_{mn}^*
m=1...6; n=1...6	$W_{mn}^*(C_{ij}, C_{kl})$
m=1...6; n=7...12	$W_{mn}^*(C_{ij}, C_{kl} + \varepsilon)$
m=7...12; n=1...6	$W_{mn}^*(C_{ij} + \varepsilon, C_{kl})$
m=7...12; n=7...12	$W_{mn}^*(C_{ij} + \varepsilon, C_{kl} + \varepsilon)$

To give an illustration on performance of the method for isotropic material, the stiffness of Mooney-Rivlin rubber model will be evaluated. The deviatoric-volumetric split of Mooney-Rivlin energy function itself is given as

$$W = c_{10}(J^{-1/3}I_1 - 1) + c_{01}(J^{-1/3}I_2 - 1) + \frac{1}{2}\kappa(J - 1)^2 \tag{12}$$

where $I_1 = \text{tr}(\mathbf{C})$ and $I_2 = \frac{1}{2}[(\text{tr} \mathbf{C})^2 - \text{tr}(\mathbf{C}\mathbf{C})]$ are the first invariant and second invariant of the right Cauchy deformation gradient, respectively. Meanwhile, the Jacobian $J = \det \mathbf{F}$ measures volume change of the material indicating incompressible and compressible deformation. In Equation (12), the material input parameters c_{10} , c_{01} , and κ , are shear and bulk constants, respectively.

4. NUMERICAL EXPERIMENTS

In order to test the LD-FEM capability simultaneously with respect to in numerical evaluation of material stiffness, a cube with a unit dimension will be used as the test specimen. The cube is meshed in Gmsh using six tetrahedral elements shown in Figure 3. The material constants for the Mooney-Rivlin model in (12) are $c_{10} = 10$ MPa, $c_{10} = 15$ MPa, and $\kappa = 10000$ MPa. While the cube is fixed in one side, the opposite side is loaded in the three different loading directions, i.e., x, y, and z, which represent tensile, bending and torsion loading, respectively. The simulations are carried out using 10 steps of load increment for each loading case.

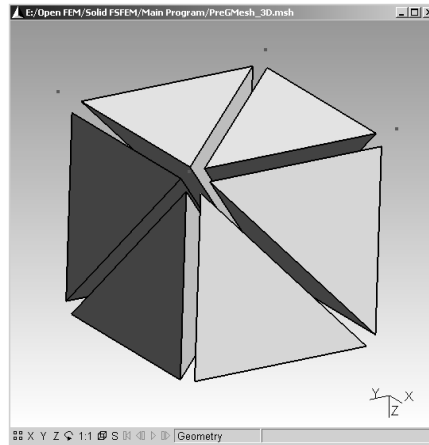


Figure 3. The unit cube with six tetrahedral elements visualized in the Gmsh

Simulation results of the cube are then processed using Gmsh (see Figure 4). In the figure, deformation compatibility across element edges are apparently observed by the same contour for both uniaxial and torsional loading cases. Hence, the compatibility proves the correctness of LD-FEM computational code with respect to the tetrahedral element formulation.

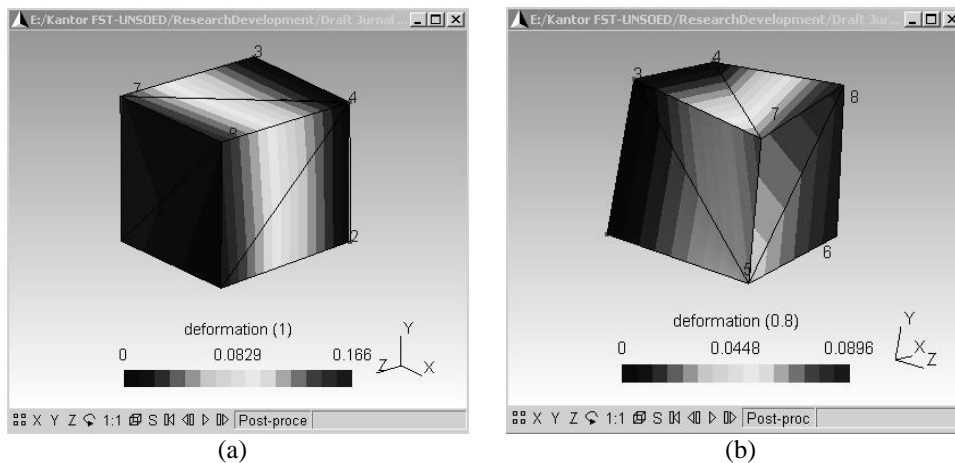


Figure 4. Deformation compatibility across element given by LD-FEM in case of (a) uniaxial loading in z-direction and (b) torsion loading with respect to z-axis by coupled forces

Meanwhile, the implementation of numerical evaluation of stiffness matrix coming from Mooney-Rivlin model shows very good results. Below a stretch of 10% differences between analytical and numerical evaluation on tangent stiffness is not lower than 2 % for each loading case as shown in Figure 5. In case of uniaxial and torsion loading the error starts growth but in a very slow rate. Thus, for newly developed isotropic material models, the numerical method expressed in (11) can be used as a validation tool for analytical evaluation of material tangent stiffness. For completeness, the future research may consider its implementation to anisotropic materials.

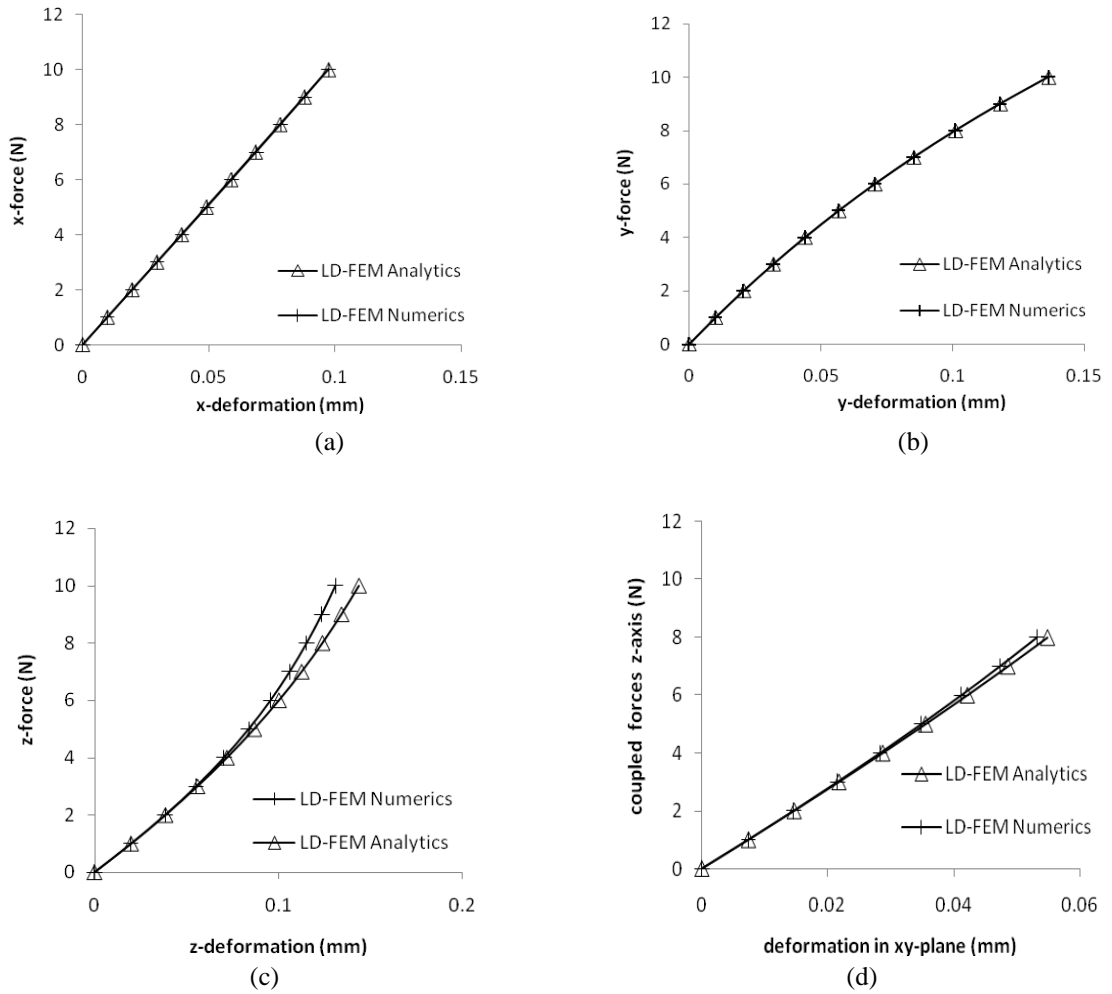


Figure 5. Response of LD-FEM after simulation at the same nodal location: shear force in x-direction (a) and y direction (b), uniaxial forces in z-direction (c) and torsion by coupled forces along z-axis (d). Nonlinear curves commonly found in large deformation are clearly observed here.

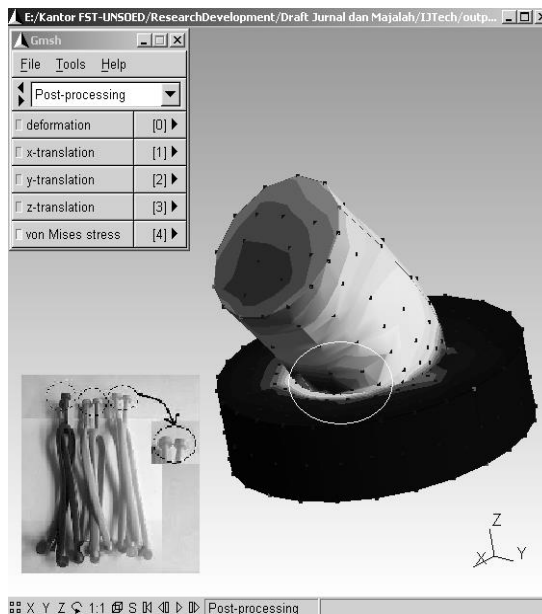


Figure 6. Modelling and simulation of rubber slipper rope using LD-FEM indicating location of stress concentration (inside white circle).

Finally, to show the LD-FEM performance in more complex problem an example of rubber slipper rope modeling and simulation is done. While using the same material parameter as the cube simulation, the rope is loaded by tensile forces in the z-direction (see Figure 6) to mimic human foot movement. Support mechanism between the rope and the body of slipper are given as nodal constraint in the opposite direction of force. The simulation result in Fig. 6 clearly indicates stress concentration at the rope corner.

5. CONCLUDING REMARKS

Based on the numerical experimental, the LD-FEM works successfully to simulate the large deformation kinematics which is commonly used to represent rubber material behaviors. It is also capable to evaluate numerical derivation of the material stiffness directly from the respective strain energy function using the Gill-Murray method. Errors indicated by comparing it to the analytical derivation for particular loading cases specified in this work, i.e., bending and torsional, are less than 2 %. Therefore, LD-FEM offers a promising capability to validate analytical evaluation of material stiffness from newly developed strain energy functions for new isotropic hyperelastic materials.

6. REFERENCES

- [1] Madutujuh, N., 1989, SANSIRO: Structural analysis software. *Engineering Software Research Center (ESRC)*, Bandung.
- [2] Alisjahbana, S. W., 1998, "Elemen Segitiga untuk Masalah Elastisitas Dua Dimensi," *Jurnal Teknik Sipil F.T. UNTAR*, No.1 Tahun ke IV.
- [3] Lumantarna, B. & Santoso, B., 2000, "Aplikasi Visual untuk Program Elemen Hingga dengan Elemen Segitiga dan Segiempat Subparametrik dan Isoparametrik," *Dimensi Teknik Sipil*, Vol.2, 2: 77-82.
- [4] Saragi, E. & Utaja., 2007, "Pembuatan program solver analisis distribusi tegangan berbasis metoda elemen hingga (MEH) pada komponen struktur," *Prosiding Pertemuan Ilmiah Nasional Rekayasa Perangkat Nuklir*, Serpong, Banten.
- [5] Dolu, H. & Hasan, H., 2010, "Metoda elemen hingga dengan program Matlab dan aplikasi SAP 2000 untuk analisis struktur cangkang," *Jurnal SMARTek*, No. 2, 8:153-168.
- [6] Wicaksono, V. A., 2011, "Pengembangan perangkat lunak menggunakan elemen hingga untuk perancangan lentur balok beton bertulang," *Undergraduate Thesis*, ITS, Surabaya.
- [7] Vaught, A., 2006, *G95 FORTRAN Compiler*, available at: <http://ftp.g95.org>, last accessed October 2, 2013.
- [8] Geuzaine, C. & Remacle, J. F., 2009, "Gmsh: a three dimensional finite element generator with built-in pre- and post-processing facilities" *International Journal for Numerical Methods in Engineering*, 0:1-24.
- [9] Bathe, K. J., 1996, *Finite element procedures*, Prentice-Hall, Upper Saddle River, NJ.
- [10] Cisloiu, R., 2006, "A stabilized mixed finite element formulation for finite strain deformation", *PhD Dissertation*, University of Pittsburgh.
- [11] Gill, P. E., Murray, W., Saunders, M. A. & Wright, M. H., 1980, "Computing the Finite Difference Approximations to Derivatives for Numerical Optimization", *Technical Report*, Department of Operations Research, Stanford University, 1980.

7. ACKNOWLEDGEMENT

I would like to thank Shankar Shrestha (formerly Research Assistant in Institute for Mechanics of Materials and Structures, Faculty of Civil Engineering, TU Vienna, Austria) for proof reading this manuscript.