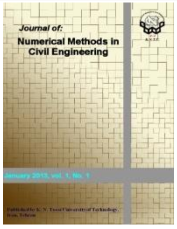


Numerical Methods in Civil Engineering

Speeding up the Stress Analysis of Hollow Circular FGM Cylinders by Parallel Finite Element Method

Saeed Asil Gharebaghi* and Negin Niknam**

ARTICLE INFO

Article history:
Received:
February 2019.
Revised:
May 2019.
Accepted:
June 2019.

Keywords:
Finite Element Method,
Parallel Processing,
Functionally Graded Material,
Speed up.

Abstract:

In this article, a parallel computer program is implemented, based on Finite Element Method, to speed up the analysis of hollow circular cylinders, made from Functionally Graded Materials (FGMs). FGMs are inhomogeneous materials, which their composition gradually varies over volume. In parallel processing, an algorithm is first divided to independent tasks, which may use individual or shared data. Such tasks could be simultaneously executed. In this paper, a parallel Finite Element software is developed to perform the analysis on a multiprocessor system. The software parallelizes every time-consuming task of the algorithm, if possible. As an application, the analysis of a thick hollow cylinder, made from FGM, is performed to evaluate the capability of the software. The results show not only the software is authoritative of analyzing large-scale problems, but also it is 2.4 times faster than the serial version. Although such speedup is achieved using eight processors, the number of processors could be increased utilizing computer networks. According to the results, it could be concluded that the speedup increases when the number of processors increases. However, because of some technical limits and overheads such as data traffic among the processors, the speedup approaches its maximum for a certain number of processors.

1. Introduction

Recent developments in materials science have encouraged researchers to use a specific type of composite materials, known as Functionally Graded Materials (FGMs). They are usually made from a composition of metal and ceramic and have specific thermo-mechanical properties. Such FGMs were firstly introduced by a group of researchers in Japan [1, 2]. FGMs are very attractive since they could be used in a wide range of engineering applications. Increasing use of FGMs in pressure vessels also induced comprehensive studies.

There are numerous numerical and analytical studies for cylindrical vessels with constant/variable thickness, made of an isotropic and non-homogeneous FGM in the literature. Horgan and Chan (1999) [3] presented an exact solution for static analysis of FGM pressure vessels and disks. Ching and Yen (2005) [4] determined stresses in FGM cylinders subjected to internal pressure by the Meshless local Petrov-Galerkin (MLPG) method. Jabbari *et al.* (2006) [5] presented a solution for thermal and mechanical stresses in hollow FGM cylinders. Arshad *et al.* (2007) [6] demonstrated the effect of material distributions on natural frequencies of FGM circular cylindrical shells. Gilhooley *et al.* (2008) [7] used MLPG method for static and dynamic analysis of orthotropic FGM rotary disks and cylinders subjected to internal pressure. Li and Peng (2009) [8] introduced a new method for stress analysis of FGM cylinders with arbitrarily varying material properties. Tutuncu and Temel (2009) [9] determined displacements and stresses in hollow FGM cylinders, disks and spheres subjected to internal pressure using plane elasticity theory and complementary functions method. Shah *et al.* (2009) [10] studied the influence of an exponential volume fraction law on the vibration frequencies of thin functionally graded cylindrical shells. Foroutan *et al.* (2011) [11] used mesh-free method for static analysis of FGM cylinders subjected to internal and external pressure. Arshad *et al.* (2011) [12]

** Corresponding Author: Assistant Professor, Department of Civil Engineering, K. N. Toosi University of Technology, Tehran, Iran. E-mail: asil@kntu.ac.ir*

*** PhD Candidate, Department of Civil Engineering, K. N. Toosi University of Technology, Tehran, Iran.*

studied the influence of various boundary conditions on vibration characteristics of thin functionally graded cylindrical shells. Asemi *et al.* (2012) [13] analyzed a thick short length hollow cylinder made from FGMs under internal impact loading. They applied Finite Element Method (FEM), based on Rayleigh-Ritz energy formulation, to study the propagation of elastic waves. Bahri *et al.* (2016) [14] developed a graded FEM code, based on Rayleigh–Ritz energy formulation. They studied the elastic behavior of a layered plate, loaded by a solid isotropic cylinder and a functionally graded interlayer. In addition, the effects of different thicknesses and exponents of power law of functionally graded interlayer on the distribution of displacements and stresses were investigated. Najibi and Shojaeefard (2016) [15] presented a new 2D-FGM material model, based on Mori-Tanaka scheme and a third-order transition function for thick hollow cylinders. They utilized FEM for elastic mechanical stress analysis of finite length cylinders. Celibi *et al.* (2017) [16] employed a novel approach to a general solution of one-dimensional steady state thermal and mechanical stresses in a hollow thick cylinder made from FGMs. Many researchers have been utilized FEM to provide a mathematically simplified procedure to model complex structures. However, such procedures are time-consuming, especially when the number of elements increases.

Since any complexity of FE model and formulation results in a bottleneck in the computation, the process required to be optimized. In light of current high-performance computing tendencies, any method intended to resolve this issue must efficiently utilize parallel computing techniques. Qian, Chengguo and Ge (2010) [17] paralleled computing of large-scale wheel/rail rolling contact FE model. Paszyński *et al.* (2010) [18] presented a new parallel multi-frontal direct solver, dedicated for the hp-FEM. Kennedya and Martins (2014) [19] developed an integrated parallel Finite Element Analysis (FEA) tool for large-scale gradient-based design optimization. It is mostly designed to use specialized parallel solution methods to solve large-scale high-fidelity structural optimization problems. Koric *et al.* (2014) [20] evaluated the performance of massively parallel direct and iterative methods for solving large sparse systems of linear equations on a high-performance computing system.

In the present work, a parallel FEM software is implemented from scratch and used for static analysis of hollow circular FGM cylinders subjected to internal pressures. Here, Young's modulus only depends on the radial coordinate, with a constant Poisson's ratio. It should be emphasized that the software is not limited to any particular choice of material. Thus, arbitrary functions that define the variation of material could be added to the source code.

2. Parallel Processing

In most of recent scientific fields, applications with high computational efforts and large data requirements have been developed. Such applications could not be executed in a traditional manner, where a command waits for all other previous commands to be accomplished. In serial computing, the Central Processing Unit (CPU) carries out just one operation at each time. Hence, the tasks are processed in sequential order. As a result, a subsequent routine starts processing only after the completion of the previous one [21]. Each process may take a long time when data is massive. Subsequently, the total required time for a practical serial application may be drastically increased. High Performance Computing (HPC) is one of the utmost important solutions to resolve the issue. Thus, there are extensive researches on HPC and parallel processing. The idea of using parallel processing method was presented in the 1950s. The method provides simultaneous processing by dividing the problem into smaller independent sections, thereby reducing process time [22]. The processors may be located in one computer or several computers, which are connected to make a network with a certain architecture. In the present research, because of the lack of such networks, all the results of the implemented software have been obtained using an eight-processor computer. However, the software is capable of being used in a network of computers, which results in a higher performance.

2.1 Speed up

The speed up of a parallel program is a simple indicator, which provides a brief overview of the amount of saved time when the program is executed in parallel. The speed up ,η, could be computed using Eqn. 1 [23].

$$\eta = \frac{t_p}{t_n}, \tag{1}$$

$$E = \frac{t_p}{t_n \times n}, \tag{2}$$

where t_p is the execution time as if the program was executed by one processor, t_n is the execution time, when the program is executed using n processors. E is the efficiency of the parallelized program. It indicates whether or not the parallelization has been accomplished successfully. The amount of parallelizable commands with respect to non-parallelizable segments in an algorithm has a profound impact on E. Any parallelization overheads through degradation in performance will decrease E so that it may even be less than one. Besides, E may be affected by network traffic, and runtime interrupts.

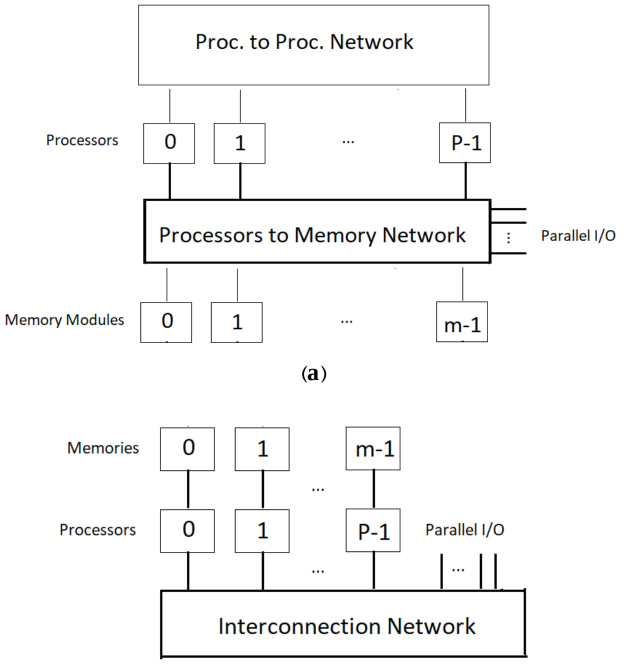
2.2 Hardware and software platforms

Parallel processing requires computer hardware, which could host more than a single processor if needed. According to the level at which the hardware supports parallelism, different computer architectures have been introduced by manufacturers. Multi-core computers that consist of multiple processing units in a single machine, and clusters which use multiple computers to work simultaneously on an algorithm are the commonly known solutions.

Memory is also an important part of a parallel platform that could be shared or distributed. The shared memory may be visualized as a central location, where all processors can access to. Fig. 1.a shows a possible hardware organization for a parallel processor with shared memory. Distributed memory architecture could be conceptually viewed as in Fig. 1.b. A collection of *p* processors, each with its own private memory, communicates through a network [24]. Technically speaking, in real situations, a software may benefit not only from shared, but also from distributed memory. Due to lower cost, systems with distributed memory are more practical. One of the hardware platforms that is commonly used for parallel processing is called cluster. A cluster is a collection of computers with a dedicated interconnection network, which operates as a unified computing system. This technology is capable of making fast and powerful systems from ordinary to general-purpose computers [25, 21].

Adapted operating system and library for writing parallel programs should be noted as the software platform. In this context, Linux-based operating systems are the best choice because they are stable, safe, and open source. Besides, there are numerous free and open source software written for Linux. In the current research, the Ubuntu operating system, which is one of the Linux distributions has been used.

Moreover, Message Passing Interface (MPI) library has been utilized to use distributed memory [23]. MPI could be used as a library for data transmission and communication among distributed memory of clustered systems. It is based on sending and receiving messages. A point-to-point communication involves one sender and one receiver processor. Additionally, all processes could be involved in a collective communication. Some of the important functions in the collective communications are `MPI_BARRIER`, `MPI_BCAST`, `MPI_GATHER` and `MPI_SCATTER`. `MPI_BARRIER` implies a synchronization point among processes. It means that all processors must reach a certain point in the source code before they could continue. In fact, if a processor has finished its task, it should wait for the others to finish their task, as well. `MPI_BARRIER` may leave negative impacts on performance because of time overheads. In MPI, `MPI_BCAST` is used when the same data should be sent from one processor to all the others in a communicator. Gathering values together from a group of processors is done by `MPI_GATHER`. Besides, `MPI_SCATTER` does the opposite task of `MPI_GATHER`. Fig. 2 illustrates how these two functions operate.



(**a**)

(b)

Fig. 1: A parallel processor with (a) shared memory, (b) distributed memory

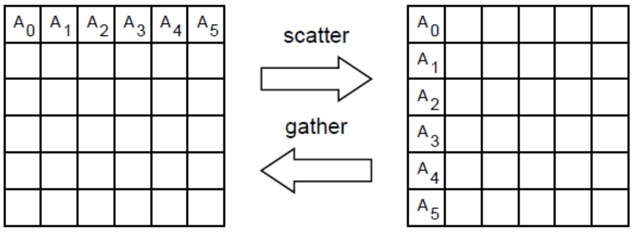


Fig. 2: The illustration of `MPI_GATHER` and `MPI_SCATTER` functions [26]

In the present work, MPI has been utilized as a software platform to implement the algorithm of parallel FEM.

3. Problem Formulation

Consider a section of a hollow circular cylinder subjected to uniform pressure on the inner and outer surfaces. Let r_i and r_o denote the inner and outer radii, and P_i and P_o be the uniform inner and outer pressures, respectively, as shown in Fig. 3. It is assumed that the body is composed of a linearly elastic inhomogeneous isotropic material with material properties that vary only in radial direction. Since the variation in Poisson's ratio is less practical than elastic modulus, one could assume that $v(r)$ is constant while $E(r)$ is a function of r. This assumption, which is commonly made in the literature of FGMs leads to a considerable mathematical simplification. In the current work, the variation of modulus of elasticity is given by a simple power material grading rule, given in Eqn. 3.

$$E(r) = E_i(r/\mathrm{r}_i)^{\beta}, \tag{3}$$

where

$$\beta = \ln(E_o/E_i)/\ln(\mathrm{r}_o/\mathrm{r}_i) \tag{4}$$

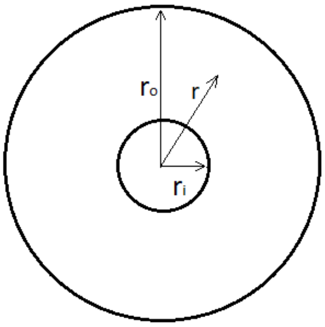


Fig. 3: The cross section of a cylindrical pressure vessel with inner radius of r_i and outer radius of r_o.

Here E_i and E_o denote the modulus of elasticity of the inner and outer surfaces, respectively [27]. It has been shown that using continuously varying properties of inhomogeneous materials in iso-parametric Finite Element formulation does not cause any computational issue [15]. However, a fine mesh of elements is required for Finite Element modelling of FGMs even for relatively small sized bodies, particularly in nonlinear analyses. Here the 3-node Constant Strain Triangular (CST) element has been used to discretize the domain. Using the principle of minimum potential energy, the elements' stiffness matrices will be formed and a system of equations is obtained as

$$[K]\{d\} = \{F\}, \tag{5}$$

where, $[K]$ is the global stiffness matrix, $\{d\}$ is the nodal displacements vector, and $\{F\}$ is the corresponding nodal force vector. The global stiffness matrix is obtained from assembling elements' stiffness matrices. Assuming a constant element thickness, the element stiffness matrix is calculated by

$$\left[k^{(e)}\right] = \mathrm{t}\int_A [B]^T[D][B]\,dA, \tag{6}$$

where, t is the thickness of e^{th} element, A is the element area, and $[B]$ is the matrix of partial derivatives of the interpolation functions, also known as the strain-displacement matrix [28].

Next, the boundary conditions are applied to Eqn. 5. Consequently, Eqn. 5 is solved, and the unknown nodal displacements are obtained. Having the nodal displacement vector, other required variables such as strain and stress could be calculated in post-processing phase. The element strain and stress values are calculated as

$$\{\varepsilon\} = [B]\{\delta^{(e)}\}, \tag{7}$$

$$\{\sigma\} = [D][B]\{\delta^{(e)}\}, \tag{8}$$

where $\delta^{(e)}$ represents the nodal displacement vector of e^{th} element and $[D]$ is the elastic material property matrix. For plane strain problems, $[D]$ could be presented as

$$[D] = \frac{E(r)}{(1+v)(1-2v)}\begin{bmatrix} 1-v & v & 0 \\ v & 1-v & 0 \\ 0 & 0 & 0.5-v \end{bmatrix}, \tag{9}$$

where, E and v are modulus of elasticity and Poisson's ratio, respectively.

4. Parallel Finite Element Analysis

The parallel Finite Element solver, as implemented in the current research, must efficiently perform three essential tasks: 1) creating and assembling the stiffness matrices of all elements, 2) solving the linear system of equations, which is created from Finite Element discretization, and 3) evaluating the stress and strain of the elements. It worth noting that each of the above-mentioned tasks is sequentially executed. However, each task is implemented in such a way that it could be executed in parallel. The most challenging task among the above-mentioned tasks is to implement an efficient parallel solver for the linear system of equations. Typically, the matrices arising from the Finite Element discretization of structures are symmetric. In the following section, the development stages of Finite Element software are presented. The software is implemented so that it could be used for FGMs.

4.1 Data distribution

To implement a software in parallel, it is essential to appropriately distribute data among processors. Thus, each processor performs its job using its own data and the program is executed in parallel as well. In the current work, all matrices and vectors have been distributed among the processors in a row-wise approach to solve the system of equations in parallel. For example, a 9 by 9 matrix will be partitioned into three rows on the first processor, three on the second and three on the third as shown in Fig. 4.

To distribute data in parallel stages of Fig. 5, the total number of elements is divided by the number of processors, so that each processor participates in the analysis by receiving a certain set of the elements. The residual elements of the division, if exist, are assigned to the first processor. Therefore, in the process of computing the stiffness matrix of each element, assembling the global stiffness matrix, and calculating the stress and strain matrices of each element, every processor performs its job using its own received data.

$$\left[\begin{array}{ccc|ccc|ccc}
1 & 0 & 4 & 11 & 3 & 0 & 0 & 1 & 0 \\
0 & 5 & 0 & 2 & 0 & 0 & 5 & 12 & 6 \\
4 & 0 & 10 & 5 & 0 & 13 & 0 & 1 & 0 \\
\hline
11 & 2 & 5 & 15 & 2 & 7 & 0 & 5 & 0 \\
3 & 0 & 0 & 2 & 6 & 4 & 2 & 0 & 8 \\
0 & 0 & 13 & 7 & 4 & 11 & 0 & 0 & 6 \\
\hline
0 & 5 & 0 & 0 & 2 & 0 & 8 & 4 & 0 \\
1 & 12 & 1 & 5 & 0 & 0 & 4 & 5 & 0 \\
0 & 6 & 0 & 0 & 8 & 6 & 0 & 0 & 9
\end{array}\right]$$

Fig. 4: An example of distributing data among three processors

4.2 Algorithm Parallelization

The main stages of the FEM analysis are presented in Fig. 5. In a serial program, solving the linear system of equations takes a great amount of time in the analysis process. Therefore, focusing on the parallelization of this task results in a considerable reduction in time. Moreover, parallelizing the element stiffness matrix calculation, the global stiffness matrix formation and the elements stress and strain calculation, will speed up the program as well. The flowchart of the program is shown in Fig. 5.

At the beginning, the software reads the properties of the problem, including material properties, mesh data, *etc*. Next, the parallelization strategy is followed by calculating the stiffness matrix of the elements and assembling the global stiffness matrix. This step is executed in parallel. It is notable that the global stiffness matrix is constructed in a compressed form. Consequently, according to the boundary conditions of the problem, the global stiffness matrix is reduced. In the next step, the resulted system of linear equations is solved for unknown nodal displacements. To this end, Generalized Minimal Residual (GMRES) method with ILU(0) preconditioner has been used in parallel. Later, the global displacement vector is formed. It is shared with processors so that each processor forms nodal displacement vectors of a certain set of elements and compute their strain and stress matrices. Eventually, other post processing computations are accomplished if needed.

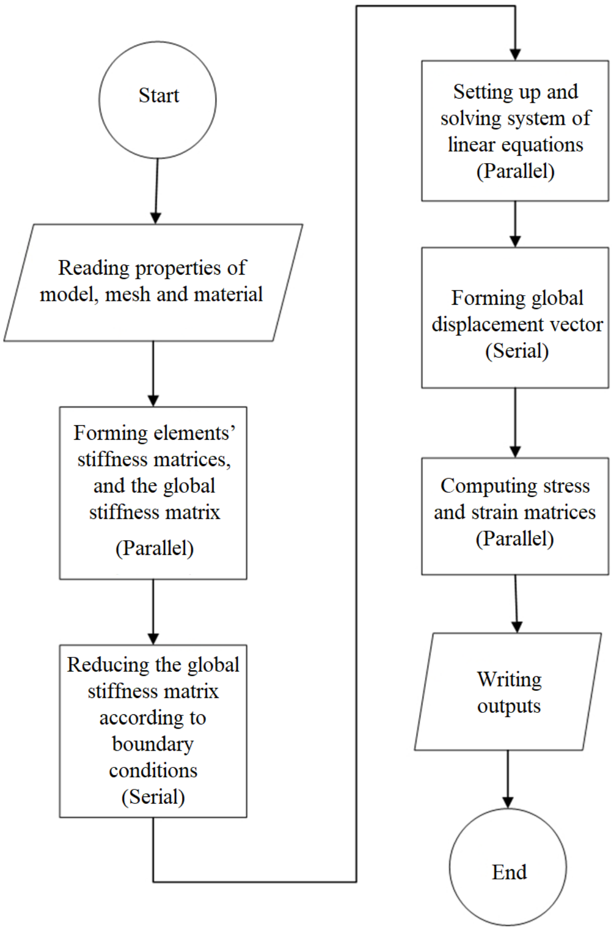

Start
Reading properties of model, mesh and material
Forming elements' stiffness matrices, and the global stiffness matrix (Parallel)
Reducing the global stiffness matrix according to boundary conditions (Serial)
Setting up and solving system of linear equations (Parallel)
Forming global displacement vector (Serial)
Computing stress and strain matrices (Parallel)
Writing outputs
End

Fig. 5: The flowchart of Finite Element Analysis

5. Verification

In this section, a cylinder subjected to unit internal pressure P is analyzed by the developed software. The model properties are presented in Table 1. The results for radial stress σ_r and hoop stress σ_θ are compared with the analytical solutions, introduced by Horgan and Chan [3], and derived analytical benchmark solutions by Tutuncu and Temel [9]. For eleven points through the thickness, the comparison results are shown in Table 2. The relative errors are calculated by $\left|\frac{Analytical\ value - FEM\ result}{Analytical\ value}\right| \times 100$ and presented in Table 3. As it can be seen, the results obtained from the software demonstrate a good agreement between numerical and analytical values. As it can be concluded from Table 4, the number of elements are intentionally increased to investigate the performance of the software.

Table 1: Model properties [9]

Material	*E (GPa)*	ν	r_o/r_i
Metal	200	0,3	2
Ceramic	360	0,3	2

[DOR: 20.1001.1.23454296.2019.3.4.3.9]

[DOI: 10.29252/nmce.3.4.1]

Table 2: Comparison of FEM results with analytical results [9]

$r_{o/}\ r_i$	σ_θ(GPa)		σ_r(GPa)	
	Analytical	FEM	Analytical	FEM
1,0	1,202558	1,202558	-1,000000	-1,000000
1,1	1,131098	1,131098	-0,803165	-0,803165
1,2	1,076507	1,076507	-0,644350	-0,644350
1,3	1,034230	1,034230	-0,513670	-0,513670
1,4	1,001160	1,001160	-0,404335	-0,404335
1,5	0,975110	0,975110	-0,311538	-0,311538
1,6	0,954510	0,954510	-0,231791	-0,231791
1,7	0,938200	0,938198	-0,162507	-0,162507
1,8	0,925311	0,925311	-0,101729	-0,101729
1,9	0,915184	0,915184	-0,047951	-0,047951
2,0	0,907305	0,907305	0,000000	0,000000

Table 3: Relative errors

$r_{o/}\ r_i$	$E_{\sigma_\theta}(\%)$	$E_{\sigma_\theta}(\%)$
1,0	1,66E-06	1,00E-06
1,1	8,84E-07	1,25E-06
1,2	1,02E-05	4,66E-05
1,3	8,70E-05	9,73E-05
1,4	2,00E-05	4,95E-06
1,5	5,13E-05	9,63E-06
1,6	3,14E-05	4,31E-05
1,7	2,11E-04	1,23E-05
1,8	8,65E-06	1,97E-05
1,9	1,42E-05	6,26E-05
2,0	6,61E-06	0,00

6. Performance and Discussion

To demonstrate the performance of the parallelized software, two different FGM cylinders have been analyzed. The material properties of metal and ceramic are presented in Table 4 for each model. It is assumed that the inner and outer surfaces are made from metal and ceramic, respectively. Only one quarter of Fig. 3 has been modeled and analyzed because of symmetry. The properties of the prepared models are presented in Table 5. The Internal pressure is assumed to be 80 and 30 MPa for the first and second model, respectively.

Table 4: Material properties [29, 15]

Model	Material	*E* *(GPa)*	ρ (kg/m^3)	ν
1	Metal (Ni)	199,5	8900	0,3
	Zirconium oxide (ZrO$_2$)	116,4	3657	0,3
2	Metal (Al)	69	2700	0,33
	Aluminum oxide (Al$_2$O$_3$)	390	3950	0,33

Substituting the given material properties in Eqs. 3-4, the variation of modulus of elasticity for the first model is determined as follows

$$\beta_1 = \frac{\ln\left(\frac{116.4}{199.5}\right)}{\ln\left(\frac{0.2}{0.1}\right)} = -0.7773, \tag{10}$$

$$E(r)_1 = 199.5(r/0.1)^{-0.7773}. \tag{11}$$

For the second model, the variation of modulus of elasticity is calculated as:

$$\beta_2 = \frac{\ln\left(\frac{390}{69}\right)}{\ln\left(\frac{0.15}{0.1}\right)} = 4.2717, \tag{12}$$

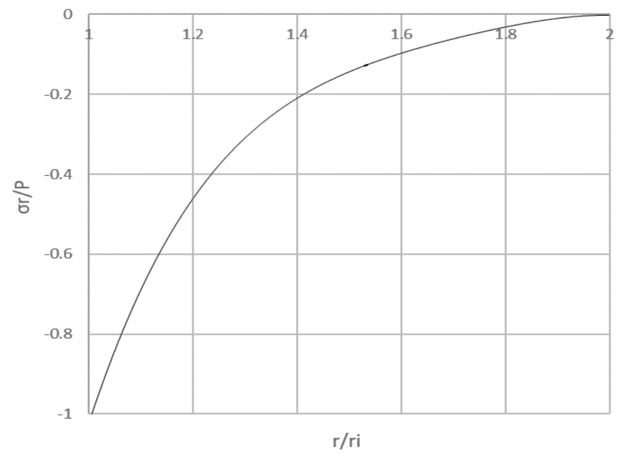
$$E(r)_2 = 69(r/0.1)^{4.2717}. \tag{13}$$

Table 5: Models properties

Model	r_i (m)	r_o (m)	T (m)	Number of Elements	Number of DOFs
1	0,1	0,2	1	153236	155258
2	0,1	0,15	1	89204	90534

The variation of the normalized value of radial and hoop stresses versus normalized radial distance are shown for both models in Fig. 6 and Fig. 7, respectively.

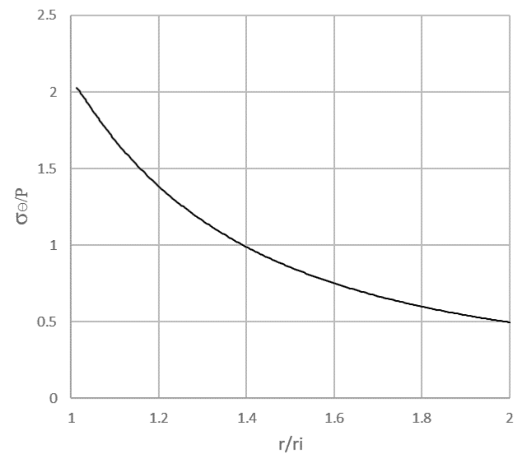
The execution times of the analyses using Constant Strain Triangular (CST) element are presented in Table 6. Measuring the execution time of each stage showed that solving the system of linear equations in parallel has consumed much more time in comparison to the other parallel stages.



(a)

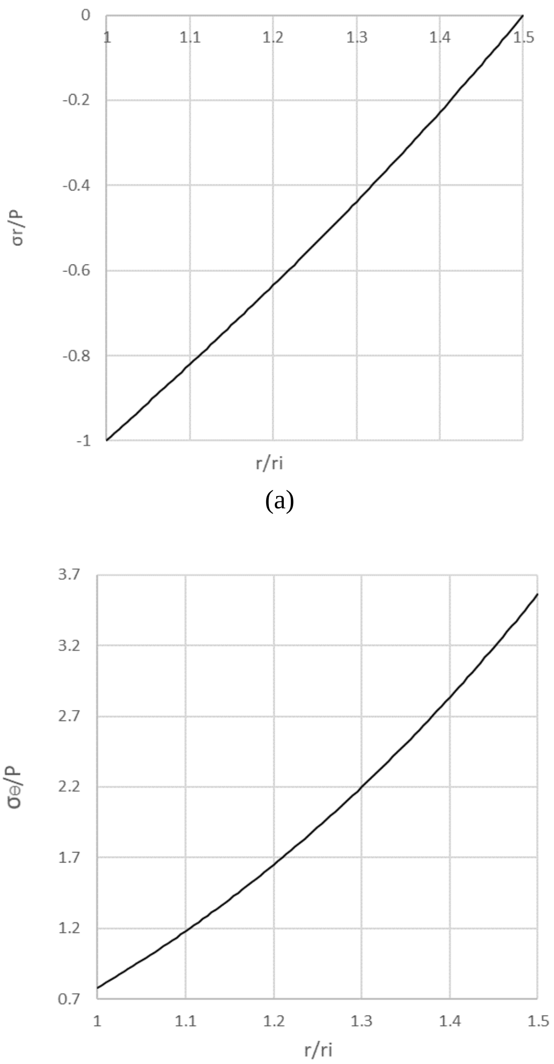
[DOR: 20.1001.1.23454296.2019.3.4.3.9]

[DOI: 10.29252/nmce.3.4.1]



(b)

Fig. 6: The variation of normalized stresses versus normalized radial distance for the first model,
(a) Radial stress, (b) Hoop stress.



(a)

(b)

Fig. 7: The variation of normalized stresses versus normalized radial distance for the second model,
(a) Radial stress, (b) Hoop stress.

Table 6: Total time of analyses (sec.)

Model	No. of Processors	1	2	3	4	5	6	7	8
1	Time (s)	1170	823	665	585	695	589	514	483
2	Time (s)	563	402	326	291	355	298	254	235

To complete the results, the speed up diagram using up to 8 processors is depicted in Fig. 8. The speed up reached to 2.42 when analyzing the first model by use of eight processors.

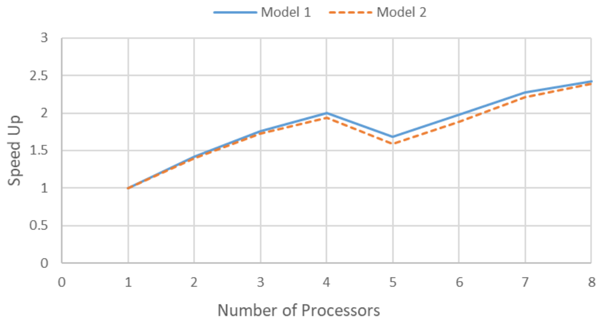


Fig. 8: Speed up of the analyses

According to the theoretical concepts of parallel processing, it is expected that by increasing the number of processors, the analysis execution time should be decreased. However, it seems that there is an issue in Fig. 8, where the number of processors is equal to five. In other words, it can be seen that the analysis time has been increased when using five processors. It is caused by the architecture of the used system that involves two CPUs, each consisting of four processors. Fig. 9 demonstrate the overview of the CPUs and their multiple levels of CPU caches. From a technical point of view, the cache is a small but fast memory storage, which is more accessible to a processor core. During program execution, the required data should be copied from the main memory to the cache. Each CPU core has two independent cache memory layers, indicated by L1 and L2. The highest level of cache memory that is shared among the cores is the third layer or L3 cache. It worth noting that, the access speed decreases from L1 to L3 layers.

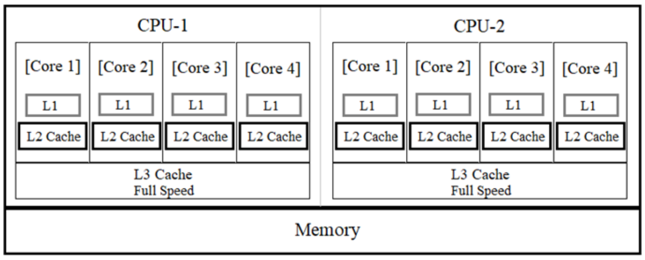


Fig. 9: An overview of CPUs and levels of CPU caches

If the software is executed with up to four processors, data is transferred via the L3 cache of CPU with a considerable access speed. However, when the fifth processor gets involved, data will be transferred through the main or Random Access Memory (RAM). Owing to the fact that the access speed of RAM is much less than L3 cache, the execution takes more time in such circumstances. Considering that computation and communication are both important in parallel processing, entering the fifth processor to the processors group results in a communication overhead. Therefore, the efficiency of parallel processing is actually reduced.

7. Conclusion

In this paper, a software was developed from scratch for parallel Finite Element Analysis (FEA) of hollow circular cylinders, made from Functionally Graded Materials (FGMs). The implementation of parallel Finite Element Method (FEM) in two-dimensional (2D) problems was described. The results demonstrated that calculating the elements' stiffness matrices, forming the global stiffness matrix, and computing the elements' stress and strain matrices could be effectively implemented in parallel. As a result, FEA spends less time to be accomplished in comparison with its serial counterpart. It is worth mentioning that increasing the number of constrained Degrees Of Freedom (DOFs) has a negative impact on the performance of the parallel software. In such situations, an increase in the number of constrained DOFs increases the execution time of forming reduced stiffness matrix, which is executed in serial. The study showed that the software achieves a notable performance for two-dimensional FEA of hollow FGM cylinders. It seems that the software is capable of being used with a large number of processors to perform FEA in parallel.

References

[1] Koizumi M. ,1993, "The concept of FGM", Ceramic Trans., Functionally Gradient Materials, 34: 3-10.

[2] Yamanouchi M., Koizumi M., Hirai T., Shiota I., 1990, "Proceeding of the first international symposium on functionally gradient materials", Japan.

[3] Horgan, C.O., Chan, A.M., 1999, "The pressurized hollow cylinder or disk problem for functionally graded isotropic linearly elastic materials", J. Elasticity, 55(1): 43-59.

[4] Ching H.K., Yen, S.C. ,2005, "Meshless local Petrov-Galerkin analysis for 2d functionally graded elastic solids under mechanical and thermal loads", J. Compos., 36(3): 223-240.

[5] Jabbari, M., Bahtui, A. and Eslami, M.R. 2006, "Axisymmetric mechanical and thermal stresses in thick long FGM cylinders", J. Therm. Stresses., 29(7): 643-663.

[6] Arshad S. H., Naeem M. N., Sultana N., 2007, "Frequency analysis of functionally graded material cylindrical shells with various volume fraction laws", Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 221: 1483–1495.

[7] Gilhooley, D.F., Xiao, J.R., Batra, R.C., McCarthy, M.A. and Gillespie, Jr.J.W., 2008, "Two-dimensional stress analysis of functionally graded solids using the MLPG method with radial basis functions" J. compu. mater. sci., 41(4): 467-481.

[8] Li X. F., Peng, X. L., 2009, "A pressurized functionally graded hollow cylinder with arbitrarily varying material properties", J. Elas., 96(1), 81-95.

[9] Tutuncu N., Temel B., 2009, "A novel approach to stress analysis of pressurized FGM cylinders, disks and spheres", Composite structures 91: 385-390.

[10] Shah A.G., Mahmood T., Naeem M.N., 2009, "Vibrations of FGM thin cylindrical shells with exponential volume fraction law", Appl. Math. Mech. -Engl. Ed. 30(5), 607–615.

[11] Foroutan M., Moradi-Dastjerdi R., Sotoodeh-Bahreini R., 2011, "Static analysis of FGM cylinders by a mesh-free method", Steel and Composite Structures 12(1): 1-11.

[12] Arshad S. H., Naeem M. N., Sultana N., Iqbal Z., Shah A.G, 2011, "Effects of exponential volume fraction law on the natural frequencies of FGM cylindrical shells under various boundary conditions", Archive of Applied Mechanics 81: 999–1016.

[13] Asemi K., Akhlaghi M., Salehi M., 2012, "Dynamic analysis of thick short length FGM cylinders", Meccanica 47(6): 1441–1453.

[14] Bahri, A., Salehi, M. and Akhlaghi, M., 2016. "Three-dimensional static and dynamic analyses of the functionally graded cylinder bonded to the laminated plate under general loading. Mechanics of Advanced Materials and Structures, 23(12): 1437-1453.

[15] Najibi A., shojaeefard M. H., 2016" Elastic mechanical stress analysis in a 2D-FGM thick finite length hollow cylinder with newly developed material model", Acta Mechanica Solida Sinica, 29(2).

[16] Celibi K., Yarimpabuc D., Keles I., 2017, "A novel approach to thermal and mechanical stresses in a FGM cylinder with exponentially varying properties", Journal of theorical and applied mechanics, 55(1):343-351.

[17] Qian X., Chengguo W., Ge G., 2010, "The research of parallel computing for large-scale finite element model of wheel/rail rolling contact", IEEE.

[18] Paszyńskia M., Pardob D., Torres-Verdínc C., Demkowiczd L., Caloe V. ,2010, "A parallel direct solver for the self-adaptive hp Finite Element Method", Journal of Parallel and Distributed Computing 70(3): 270-281.

[19] Graeme J. Kennedya, Joaquim R.R.A.Martinsb, 2014, "A parallel finite-element framework for large-scale gradient-based design optimization of high-performance structures", Finite Elements in Analysis and Design 87:56-73.

[20] Koric S., Qiyue L., Guleryuz E., 2014, "Evaluation of massively parallel linear sparse solvers on unstructured finite element meshes", Computers & Structures, V. 141: 19–25.

[21] Townsend J. T., 1990, "Serial vs. Parallel Processing: Sometimes They Look like Tweedledum and Tweedledee but They Can (And Should) be Distinguished", Psychological Science, 1(1): 46-54.

[DOR: 20.1001.1.23454296.2019.3.4.3.9]

[DOI: 10.29252/nmce.3.4.1]

[22] Ruede U., Gmeiner B., 2012, “Parallel Iterative Algorithms and Performance- Analysis and Tuning of Parallel Programs”, Vatsal Sharan, Indian Institute Of Technology Kanpur.

[23] Gropp W., Lusk E., Doss N., Skjellum A., 1996, “MPICH: A high-performance, portable implementation of the MPI message passing interface standard”. Parallel Computer 22(6):789–828.

[24] Parhami B. ,2002, “Introduction to Parallel Processing Algorithms and Architectures”, Kluwer Academic Publisher.

[25] Asil Gharebaghi S., Zanganevar M.,2012, "Application of Parallel Processing in Civil Engineering", Master of Science Seminar, K. N. Toosi University of Technology, Tehran, Iran.

[26] “MPI: A Message-Passing Interface Standard”, Version 2.2 (2009), Message Passing Interface Forum, University of Tennessee.

[27] Li C., Zou Z. and Duan Z, 2000, “Multiple iso-parametric finite element method for non-homogeneous media”, Mechanics Research Communications, 27(2): 137-142.

[28] Hutton David H., 2004, “Fundamentals of Finite Element Analysis”, McGrow-Hill.

[29] Yıldırım V., Boğa C., 2016, “Closed-Form elasticity solutions to uniform rotating discs made of a radially functionally graded material”, IJIRSET 5(12).