J. KSIAM Vol.21, No.3, 125–142, 2017 http://dx.doi.org/10.12941/jksiam.2017.21.125

DEVELOPMENT OF AN IMPROVED THREE-DIMENSIONAL STATIC AND DYNAMIC STRUCTURAL ANALYSIS BASED ON FETI-LOCAL METHOD WITH PENALTY TERM

SEIL KIM[1], HYUNSHIG JOO[1], HAESEONG CHO[2], AND SANGJOON SHIN[3†]

[1] GRADUATE STUDENT, DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING, SEOUL NATIONAL UNIVERSITY, 1 GWANAK-RO, GWANAK-GU, SEOUL, 151-744, KOREA
E-mail address: {seill49kim,joohyunshig}@snu.ac.kr

[2] BK21 PLUS TRANSFORMATIVE TRAINING PROGRAM FOR CREATIVE MECHANICAL AND AEROSPACE ENGINEERS, INSTITUTE OF ADVANCED MACHINES AND DESIGN, SEOUL NATIONAL UNIVERSITY, 1 GWANAK-RO GWANAK-GU, SEOUL, 151-744, KOREA
E-mail address: nicejjo@snu.ac.kr

[3] PROFESSOR, DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING, INSTITUTE OF ADVANCED AEROSPACE TECHNOLOGY, 1 GWANAK-RO, GWANAK-GU, SEOUL, 151-744, KOREA
E-mail address: ssjoon@snu.ac.kr

ABSTRACT. In this paper, development of the three-dimensional structural analysis is performed by applying FETI-local method. In the FETI-local method, the penalty term is added as a preconditioner. The OPT-DKT shell element is used in the present structural analysis. Newmark-β method is employed to conduct the dynamic analysis. The three-dimensional FETI-local static structural analysis is conducted. The contour and the displacement of the results are compared following the different number of sub-domains. The computational time and memory usage are compared with respect to the number of CPUs used. The three-dimensional dynamic structural analysis is conducted while applying FETI-local method. The present results show appropriate scalability in terms of the computational time and memory usage. It is expected to improve the computational efficiency by combining the advantages of the original FETI method, i.e., FETI-mixed using the mixed local-global Lagrange multiplier.

1. INTRODUCTION

Recently, the structural components used in the industrial fields have become more complicated and been required to alleviate the computational cost for the structural analysis. For the structural analysis of a large-sized structure, which has an enormous number of degrees of freedom, difficulties in terms of computational time and memory handling were generated.

Received by the editors June 29 2017; Accepted September 14 2017; Published online September 15 2017.
2000 *Mathematics Subject Classification.* 65Y05, 74K25.
Key words and phrases. FETI-local, Penalty Term, Three-Dimensional Static and Dynamic Analysis, OPT-DKT Shell Element.
† Corresponding author.

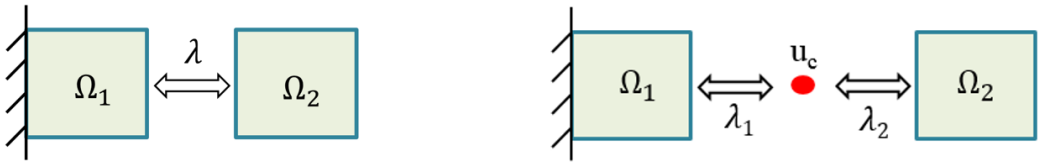
One of the most advanced approaches to alleviate such cost was the finite element tearing and interconnecting method (FETI). The basic idea of FETI method was to decompose the relevant domain into non-overlapping sub-domains and each of these was assigned to an individual CPU. And Lagrange multipliers were used to enforce the values of the degrees of freedom to coincide on the interfaces among the sub-domains [1]. The original FETI method was applied to a parallel computation algorithm for the second order partial differential equations (PDEs) [2].

Farhat [2] proposed the original FETI method as a parallel finite element computational method. He solved the local singularity problems of the floating sub-domains by using a pseudo inverse matrix. The original FETI method was extended to the dual-primal FETI (FETI-DP) method, which took standard preconditioned conjugate gradient algorithm (PCG), which was not used in the original FETI method. Due to the singularity of the stiffness matrix under the floating sub-domain, the original FETI method procedure was not used in the same way for application to static and transient dynamic analysis. On the other hand, the FETI-DP method was applicable to both static and dynamic problems [3]. Park [4] suggested alternative constraints modeling method to enforce the continuity of the displacement field by using the localized Lagrange multipliers. The local Lagrange multiplier formulations were set about the derivations of the partitioned equilibrium equations for structure. Bauchau [5-6] proposed an augmented Lagrangian formulation (ALF) with the use of global and local Lagrange multipliers. The application of augmented Lagrangian terms was shown to improve conditioning of the flexibility matrix, and thereby improving the convergence rate of the iterative procedure used to solve interface problem. In addition, he reported advanced accuracy by adopting localized Lagrange multipliers as penalty formulations [5].

In this paper the three-dimensional structural analysis by applying FETI-local method will be developed. The OPT-DKT shell element will be used in the structural analysis. Newmark-β method will be employed to extend the dynamic analysis. The three-dimensional FETI-local static structural analysis will be conducted. The computational time and memory usage will be compared with respect to the number of CPUs used. The displacement results will be compared while using different number of the sub-domains. The three-dimensional dynamic structural analysis will be conducted. Finally, the computational time in terms of the number of CPUs will be evaluated.

2. FORMULATIONS

2.1. **The FETI-local algorithm.** In the FETI-local method, the given domain is divided into non-overlapping sub-domains and each sub-domain is computed by a single CPU. The additional interface nodes are used to define localized displacement field. Each sub-domain and additional interface node are connected by local Lagrange multipliers. The FETI-local method is a combination of the localized Lagrange multiplier technique with the augmented Lagrangian formulation and is ideally reasonable for the large size structural analysis. This method uses a weighting penalty as a preconditioner unlike the original FETI method. This new approach provides an ideal preconditioning of the flexibility matrix [7].



(a) Classical Lagrange multipliers (b) Localized Lagrange multipliers

FIGURE 1. Comparison between classical and localized Lagrange multipliers

In the original FETI method, continuity of the displacement was determined by Lagrange multipliers directly as Fig. 1. (a). The constraint potential and constraint condition were defined as follows:

$$V_c = \lambda^T C \tag{2.1}$$

$$C = u_1 - u_2 \tag{2.2}$$

On the other hand, in the FETI-local method, continuity in the displacement field is enforced by imposing additional interface nodes and local Lagrange multipliers as Fig. 1. (b). The boundary nodes in each sub-domain are localized with additional interface nodes, which are used to construct constraint condition. The constraint potential and constraint condition are defined as follows:

$$V_c = \lambda^{(1)T} C^{(1)} + \lambda^{(2)T} C^{(2)} \tag{2.3}$$

$$C^{(2.1)} = u_b^{(1)} - u_c^{(1)}, \quad C^{(2)} = u_b^{(2)} - u_c^{(2)} \tag{2.4}$$

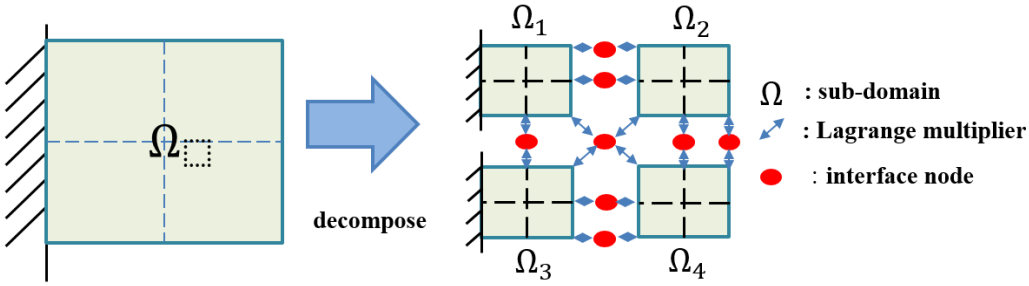


FIGURE 2. Schematic description of FETI-local method

Figure 2 shows the configuration of the connected sub-domains, through localized Lagrange multipliers. The number of additional interface nodes $\underline{\lambda}^{(i)T}$ is added to the number of internal nodes $\underline{u}^{(i)T}$ at each sub-domain. Hence, the total number of nodes for each subdomain is defined as follows:

$$\underline{\hat{u}}^{(i)} = \left\{ \underline{u}^{(i)^T} \ \underline{\lambda}^{(i)^T} \right\} \tag{2.5}$$

The total potential energy of the object structure can be decomposed into three components:

$$\Pi = A + \Phi + V_c \tag{2.6}$$

Π is the total potential energy of the entire structure, A is the strain energy of the total number of subdomains and defined as follows:

$$A = \frac{1}{2} \sum_{i=1}^{N_s} \underline{\hat{u}}^{(i)T} \underline{\underline{\hat{K}}}_{ii} \underline{\hat{u}}^{(i)} \tag{2.7}$$

where $\underline{\underline{\hat{K}}}_{ii} = \begin{bmatrix} diag\left(\underline{\underline{K}}_{\alpha\alpha}\right) & 0 \\ 0 & 0 \end{bmatrix}$. $\underline{\underline{K}}_{\alpha\alpha}$ is the stiffness matrix for the i^{th}sub-domain.

Φ is the total potential of the external loads for each sub-domain and defined as follows:

$$\Phi = -\sum_{i=1}^{N_s} \hat{u}^{(i)} \hat{Q}^{(i)} \tag{2.8}$$

where $\hat{Q}^{(i)T} = \left\{ Q^{(i)T}, \ 0 \right\}$. $\hat{Q}^{(i)T}$ is the external load for the i^{th} sub-domain.

And V_c is the potential of constraints defined by localized Lagrange multipliers.

$$\underline{C}^{(j)} = \underline{u}_b^{(j)} - \underline{u}_c^{(j)} \tag{2.9}$$

$$V_c^{(j)} = p\lambda^{(j)T} C^{(j)} + \frac{p}{2} C^{(j)T} C^{(j)} \tag{2.10}$$

$\underline{u}_b^{(j)}$is the boundary nodes with j^{th} position put on additional interface nodes. $\lambda^{(j)}$is the localized Lagrange multipliers used to enforce the local constraints.

In the original FETI method, the interface problem was solved by an iterative method. The preconditioned conjugate gradient (PCG) algorithm was used for the solution of iteration problem. Schur complement method was applied for more efficient iteration. However, the additional Lagrange multipliers were needed at the cross point for the stable PCG iteration and reasonably not scalable for fourth-order plate and shell problems [3].

On the other hand, in the FETI-local method, the penalty method is used. The penalty term p is the scaling factor for the local Lagrange multiplier. The second term of Eq. (2.10) is characteristic of the penalty method. The penalty term p plays a role as stiffness of a spring placed between the sub-domains, and Lagrange multipliers are reaction forces that impose the continuity of the displacement field along the boundary nodes. The penalty method yields an exact solution if the penalty tends to infinity, but otherwise permits certain violations of the constraint that the interpenetration has to be zero. It is necessary to estimate the magnitude of the penalty parameter to limit the penetration [8]. Finally, the $\underline{u}_c^{(j)}$can be obtained by adding the penalty term for problem with equality constraints in Eq. (2.10). Further details are expressed in Ref. 10.

To satisfy the array in Eq. (2.5), the generalized stiffness matrix and force vector can be decomposed into the boundary nodes and its additional interface nodes at each sub-domain. Additionally, $\underline{f}^{(j)}$ is the force vector in the j-th interconnecting point at an arbitrary sub-domain, and $\underline{\underline{k}}^{(j)}$ is the stiffness matrix of the boundary nodes [1].

$$\underline{f}^{(j)} = \left\{ \begin{array}{l} \underline{f_b}^{(j)} \\ X\underline{f_c}^{(j)} \end{array} \right\} = \left\{ \begin{array}{l} \left\{ p\underline{\lambda}^{(j)} + p\underline{C}^{(j)},\, p\underline{C}^{(j)} \right\}^T \\ \left\{ -p\underline{\lambda}^{(j)} - p\underline{C}^{(j)} \right\} \end{array} \right\} \tag{2.11}$$

$$\underline{\underline{K}}^{(j)} = \left[\begin{array}{ll} \underline{\underline{K}}_{bb}^{(j)} & \underline{\underline{K}}_{bc}^{(j)} \\ \underline{\underline{K}}_{cb}^{(j)} & \underline{\underline{K}}_{cc}^{(j)} \end{array} \right] = \left[\begin{array}{lll} p\underline{\underline{I}} & p\underline{\underline{I}} & -p\underline{\underline{I}} \\ p\underline{\underline{I}} & 0 & -p\underline{\underline{I}} \\ -p\underline{\underline{I}} & -p\underline{\underline{I}} & p\underline{\underline{I}} \end{array} \right] \tag{2.12}$$

To assemble both the force vector and stiffness matrix into each sub-domain, the Boolean matrix $\underline{\underline{B}}_b^{(j)}$is used to connect the sub-domain boundary nodes directly to the global nodes. Index $(.)_b$ and $(.)_c$ denote DOFs related to boundary and interface nodes, respectively [1]. More information about the Boolean matrices is provided in [9].

Finally, the governing equation of the structure could be defined as follows:

$$\left[\begin{array}{ll} \sum_{i=1}^{N_s} (\underline{\underline{\hat{K}}}^{(i)} + \underline{\underline{K}}_{bb}^{(i)}) & \sum_{i=1}^{N_s} \underline{\underline{K}}_{bc}^{(i)} \\ \sum_{i=1}^{N_s} \underline{\underline{K}}_{bc}^{(i)T} & \sum_{i=1}^{N_s} \underline{\underline{K}}_{cc}^{(i)} \end{array} \right] \left\{ \begin{array}{l} \hat{\underline{u}} \\ \underline{c} \end{array} \right\} = \left\{ \begin{array}{l} \hat{\underline{Q}} - \hat{\underline{f}}_b \\ -\underline{f}_c \end{array} \right\} \tag{2.13}$$

Three computational algorithms are utilized to solve Eq. (2.13).

In the first step, a temporary stiffness matrix and force vector are created.

$$\left[\begin{array}{ll} \underline{\underline{K}}_{11} & \underline{\underline{K}}_{12} \\ \underline{\underline{K}}_{21} & \underline{\underline{K}}_{22} \end{array} \right] \left\{ \begin{array}{l} \underline{u} \\ \underline{u}_c \end{array} \right\} = \left\{ \begin{array}{l} \hat{\underline{Q}} - \hat{\underline{f}}_b \\ -\underline{f}_b \end{array} \right\} \tag{2.14}$$

where $\underline{\underline{K}}_{11}^{(i)} = \sum_{j=1}^{N_b^{(i)}} \left(\underline{\underline{B}}_e^{(j)T} \underline{\underline{D}}_e^{(j)} \underline{\underline{B}}_e^{(j)} + p\underline{\underline{B}}_b^{(j)} \right)$, $\underline{\underline{K}}_{12}^{(i)} = \sum_{j=1}^{N_b^{(i)}} p\underline{\underline{B}}_b^{(j)T}$, $\underline{\underline{K}}_{21}^{(i)} = \sum_{j=1}^{N_b^{(i)}} p\underline{\underline{B}}_b^{(j)}$

The localized Lagrange multipliers are evaluated in the second step.

$$\underline{u}_c = \underline{\underline{K^*}}^{-1} \underline{f^*} \tag{2.15}$$

where $\underline{\underline{K^*}} = \underline{\underline{K}}_{22} - \underline{\underline{K}}_{21}\underline{\underline{K}}_{11}^{-1}\underline{\underline{K}}_{12}$, $\underline{f^*} = -f_b - \underline{\underline{K}}_{21}\underline{\underline{K}}_{11}^{-1}(\hat{\underline{Q}} - \hat{\underline{f}}_b)$

Finally, the displacement of the total number of degrees of freedom is derived.

$$\hat{\underline{u}} = \underline{\underline{K}}_{11}^{-1} \left\{ \hat{\underline{Q}} - \hat{\underline{F}}_b - \underline{\underline{K}}_{12}\underline{u}_c \right\} \tag{2.16}$$

Figure 3 shows the three steps for the present parallel computation algorithm. At first, the full geometry is decomposed into the number of sub-domains by Message Passing Interface (MPI). In Step I, Boolean matrices are defined to connect boundary nodes to interface nodes, and define the stiffness matrix and force vector in each sub-domain. In this procedure, each sub-domain is calculated in each CPU. To solve the interface problems, the assembling of the stiffness matrix and force vectors of each sub-domain into the full stiffness matrix and

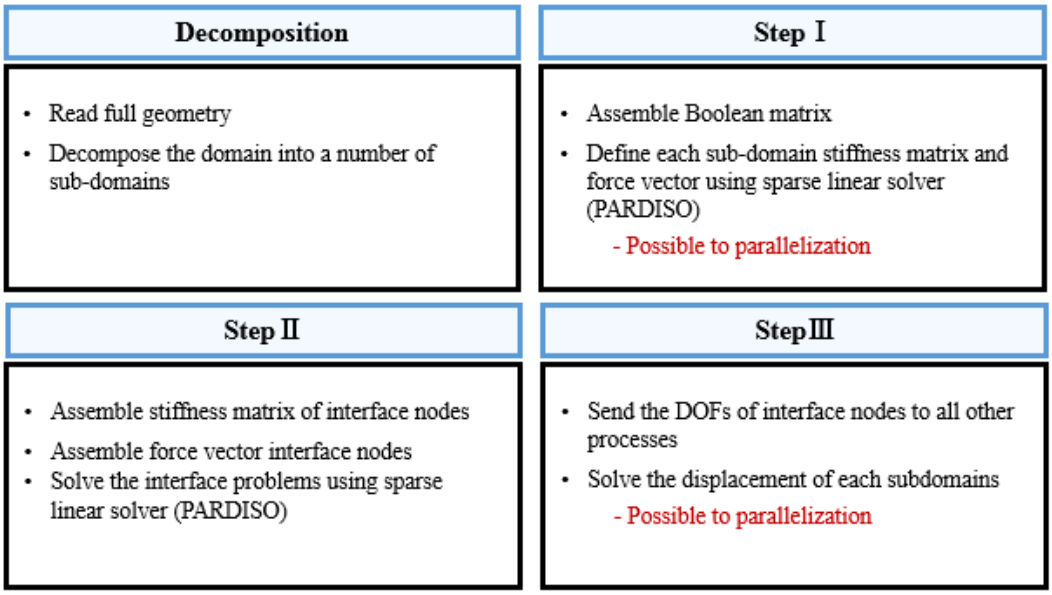


FIGURE 3. Three steps in the present parallel computing algorithm

force vector, is required. Finally, all the degrees of freedoms of interface nodes are used to solve the displacement of each sub-domain. In the third step, it is possible to solve all the displacement problems in each CPU. Displacement of the structure is solved by MPI_BCAST. A sparse matrix library called PARDISO is used in every single CPU to handle the sparse stiffness matrices. In Step, the computation of the stiffness matrix and the force vector at each sub-domain are conducted by using Eq. (2.14). The localized Lagrange multipliers and displacement of additional interface nodes are evaluated in Step by using Eq. (2.15). Finally, the displacement of the total number of degrees of freedom is derived in Step by the Eq. (2.16). In Step and Step, the computational time is related to the size of stiffness matrix of the entire system. The size of stiffness matrix is proportional to the total number of degrees of freedom including both subdomain and Lagrange multipliers. Specifically, the size of stiffness matrix of subdomain decrease in proportion to the amount of decomposed subdomain, i.e., $\underline{\underline{K}}_{11}^{(i)}$. In contrast, the size of Boolean matrix, i.e., $\underline{\underline{K}}_{12}^{(i)}$, $\underline{\underline{K}}_{21}^{(i)}$and $\underline{\underline{K}}_{22}^{(i)}$, increases in proportion to the number of local Lagrange multipliers to enhance the compatibility of displacement at each subdomain in Eq. (2.14). In hence, the number of decomposed subdomain and its relevant local Lagrange multipliers must be carefully considered.

2.2. **Formulation of the dynamic analysis.** For the dynamic analysis, the following governing equations are used.

$$M\ddot{q}^{t+\Delta t} + K^{t+\Delta t}q^{t+\Delta t} = F^{ext^{t+\Delta t}} \tag{2.17}$$

The mass matrix is derived from an element shape function ($\underline{\underline{N}}$) in the two- and three-dimensional finite element:

$$\underline{\underline{M}} = \left[\int_V \underline{\underline{N_s}}^T \underline{\underline{N_s}} dV \right] \tag{2.18}$$

The effective stiffness matrix ($\underline{\underline{K}}_{11}$)and load vector ($\hat{Q}$) in each sub-domain are

$$\underline{\underline{K}}_{11} = \frac{4}{\Delta t^2}\underline{\underline{M}} + \underline{\underline{K}} \tag{2.19}$$

$$\hat{Q} = \underline{\underline{M}} \left(\frac{4}{\Delta t^2}\underline{\ddot{u}} + \frac{4}{\Delta t^2}\underline{\dot{u}} + \underline{u} \right) + \underline{f}_{ext} \tag{2.20}$$

Newmark-β method is used to extend these for time transient dynamic analysis. It is widely used in the dynamic response of structures. Newmark-β method is more versatile than the central difference method and stable in terms of accuracy for the linear structural dynamics. Additionally, it is simple to implement to the analysis.

The first-order time derivative is solved as follows:

$$\underline{\dot{q}}_{n+1} = \underline{\dot{q}}_n + (1-\gamma)\Delta t \underline{\ddot{q}}_n + \gamma \Delta t \underline{\ddot{q}}_{n+1} \tag{2.21}$$

Since the acceleration is proportional to time, the extended mean value formulation should be extended to the second time derivative to obtain the correct displacement [10].

$$\underline{q}_{n+1} = \underline{q}_n + \Delta t \underline{\dot{q}}_n + \frac{1}{2}\Delta t^2 \underline{\ddot{q}}_\beta \tag{2.22}$$

Then, the followings are derived:

$$\underline{\ddot{q}}_\beta = (1-2\beta)\underline{\ddot{q}}_n + 2\beta \underline{\ddot{q}}_{n+1} \quad 0 \le \beta \le 1 \tag{2.23}$$

Newmark-βmethod shows a reasonable value of $\gamma = 0.5$ [10].

The displacement, velocity, acceleration, and external load vectors are $\underline{q}$, $\underline{\dot{q}}$, $\underline{\ddot{q}}$ and $\underline{f}_{ext}$, respectively. For the constant average method, $\beta = 1/4$ [10].

$$\underline{q}_{n+1} = \underline{q}_n + \Delta t \underline{\dot{q}}_n + \frac{1-2\beta}{2}\Delta t^2 \underline{\ddot{q}}_n + \beta \Delta t^2 \underline{\ddot{q}}_{n+1} \tag{2.24}$$

$$\underline{\dot{q}}_{n+1} = \underline{\dot{q}}_n + \frac{\Delta t}{2}\left(\underline{\ddot{u}}_n + \underline{\ddot{u}}_{n+1} \right) \tag{2.25}$$

$$\underline{\ddot{q}}_{n+1} = \frac{4}{\Delta t}\left(\underline{u}_{n+1} - \underline{u}_n - \Delta t \underline{\dot{u}}_n \right) - \underline{\ddot{u}}_n \tag{2.26}$$

2.3. **Three-dimensional OPT-DKT shell element.** In the three-dimensional structural analysis with parallel computation, the OPT-DKT shell element, which combines the optimal triangle (OPT) membrane element and discrete Kirchhoff plate (DKT) bending element is used. Felippa [11] developed an optimal triangle element. Each element has 9 degrees of freedom with a drilling degree of freedom. Batoz et al. [12] showed that the discrete Kirchhoff triangle (DKT) would be the most reliable element for the analysis of thin plates. Each element has 9 degrees of freedom with a bending degree of freedom. Khosravi [13] developed a new three-node triangular shell element OPT membrane element and DKT plate bending element. Each element has 18 degrees of freedom (3 translations and 3 rotation at each node) at each element. The configuration of the OPT-DKT shell element is shown in Fig. 4.

The nodal displacement vector of the OPT-DKT shell element is represented by $\{\underline{d}_{OPT-DKT}\}$.

$$\{\underline{d}_{OPT-DKT}\} = \{u_1\ \nu_1\,\omega_1\,\theta_{x1}\,\theta_{y1}\,\theta_{z1}\,u_2\ \nu_2\,\omega_2\,\theta_{x2}\,\theta_{y2}\,\theta_{z2}\cdots\theta_{z3}\}^T \tag{2.27}$$

The stiffness matrix of the shell element corresponding to the displacement vector $\{\underline{d}_{OPT-DKT}\}$ can be described as follows:

$$\underline{\underline{K}} = \int_A \underline{\underline{B_\circ^T D_\circ B_\circ}} dA = \begin{bmatrix} \underline{\underline{K}}_m & \int \underline{\underline{B_m^T D^e B_b}} dA \\ \int \underline{\underline{B_b^T D^e B_m}} dA & \underline{\underline{K}}_b \end{bmatrix} \tag{2.28}$$

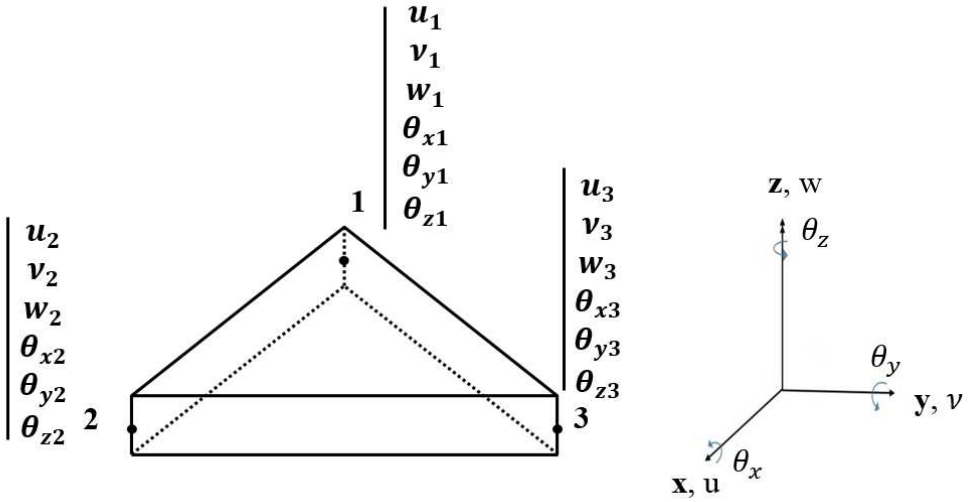


FIGURE 4. Configuration of the OPT-DKT shell element

3. NUMERICAL RESULTS

3.1. **FETI-local static analysis of a cylinder.** The three-dimensional FETI-local static structural analysis is implemented in a parallel computing hardware, using a message-passing interface (MPI). The Table 1 is the details of the present multi-core environment specifications.

TABLE 1. Specifications of the present multi-core environments

Processor	Intel® Xeon® E5-2420
Number of nodes	4
Number of CPU	50
Memory	24.6 GB
OS	CentOS 6.5
Memory transfer per second	1GB/s

Figure 5 shows the configuration of the three-dimensional static problem.

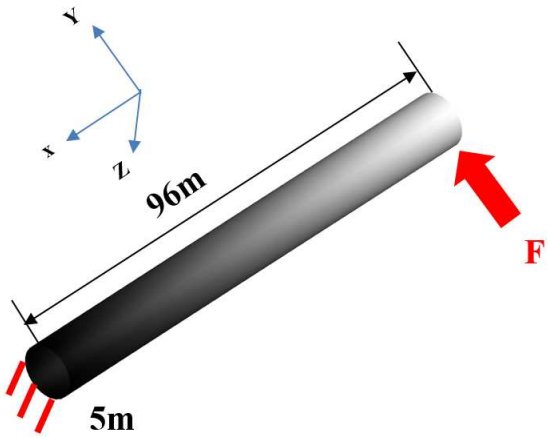


FIGURE 5. 3-D static FETI-local analysis

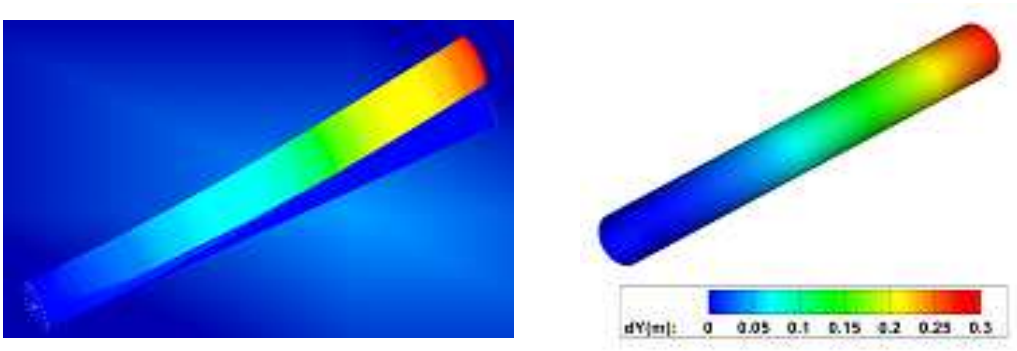
The key parameters of the analysis conditions are summarized in Table. 2. The number of total degrees of freedom is kept to 55,440, whereas the number of CPUs increases from 4 to 48. To validate the present three-dimensional FETI-local static analysis, the present result is compared with those by using NASTRAN. The Table. 3 is the comparison of the tip displacement with NASTRAN. The present result is in good agreement with that by NASTRAN, within a difference as small as 0.9%.

The three-dimensional FETI-local structural analysis is validated in the different number of the sub-domains. The contour and the displacement are compared. Figure 7 shows the comparison of the displacement in vertical-direction in terms of the number of the sub-domains. For each different number of sub-domains, the results show the same contour and displacement.

TABLE 2. Parameters for the FETI-local static analysis

Classification	Value
Young's modulus [GPa]	210
Thickness [m]	0.2
Poisson's ratio	0.3
Density	7850
Total tip loads [N]	144×10^5
Total element	18,432
Total nodes	9,240
Total degrees of freedom	55,440

Therefore, the present three-dimensional FETI-local structural analysis gives the same result regardless of the number of the sub-domains.



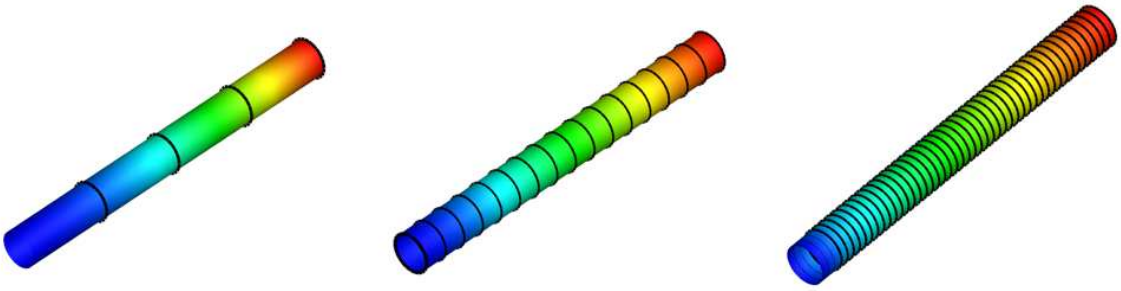
(a) Displacement by NASTRAN analysis (b) Displacement by the present

FIGURE 6. Comparison of the displacement in y-direction

TABLE 3. Tip displacement comparison with NASTRAN

External load	NASTRAN	Present	Discrepancy
1,000KN	0.2152m	0.2154m	0.11%
4,000KN	0.86m	0.868m	0.87%

Figures 8 and 9 show the trend of the computational time and memory usage while increasing the number of CPUs used from 4 to 48. However, the computational time and memory increase further when more than 48 CPUs are used. This is due to the increased degrees of freedom as defined in Eq. (2.16). As the number of CPUs increases, the degrees of freedom for the localized Lagrange multipliers will also increase. In terms of the stiffness matrix of the entire structure, the size of the matrix is proportional to the number of degrees of freedom.



(a) Displacement with 4 sub-domains

(b) Displacement with 16 sub-domains

(c) Displacement with 48 sub-domains

FIGURE 7. Comparison of the displacement in vertical-direction

Specifically, the computational time in Step increases when over 48 CPUs are used. This is because the size of the Boolean matrices, i.e., $\underline{\underline{K}}_{12}^{(i)}$, $\underline{\underline{K}}_{21}^{(i)}$ and $\underline{\underline{K}}_{22}^{(i)}$, which will increase in proportion to the number of local Lagrange multipliers in Eq. (2.14). Hence, the stiffness matrix of proportionally increased size at each decomposed sub-domain and localized Lagrange multipliers may increase the total computational time. Therefore, the number of CPUs are needed to be carefully chosen to be as small as possible for efficient parallel computation. It shows a 96.6% computational time and memory usage reduction by using 32 CPUs, when compared with those obtained by using 4 CPUs. Therefore, the present FETI-local method is efficient in the reduction of computational time and its memory usage in the three-dimensional static analysis.

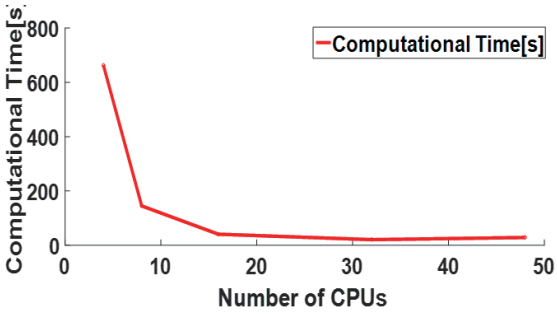


FIGURE 8. Scalability trend

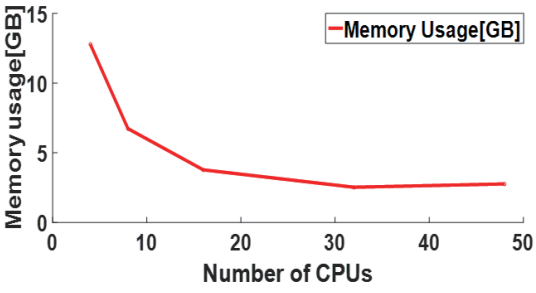


FIGURE 9. Memory usage trend

3.2. **FETI-local static analysis of a launch vehicle engine nozzle.** Figure 10 shows the configuration of an engine nozzle, discretized by 120,300 degrees of freedom, i.e., 20,050 nodes and 40,000 OPT-DKT shell elements.

TABLE 4. Computational time consumed and percentage distribution

CPUs	4		8		16		32		48	
	Time (s)	Percent (%)	Time (s)	Percent (%)	Time (s)	Percent (%)	Time (s)	Percent (%)	Time (s)	Percent (%)
Step I	625	94.3	133	91.6	35.71	87.5	17.4	83	24.7	86
Step II	0	0	0.007	0.01	0.23	0.6	0.5	2.4	0.8	3
Step III	38	5.7	12	8.39	4.8	11.9	3	14.6	3	11
Total	663		145		40.7		20.9		28.5	

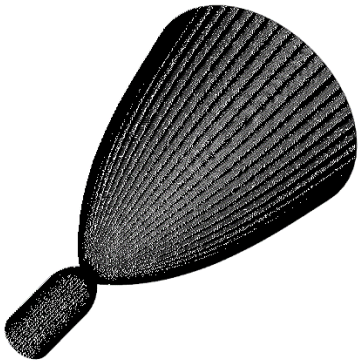


FIGURE 10. Discretized engine nozzle with 120,300 degrees of freedom

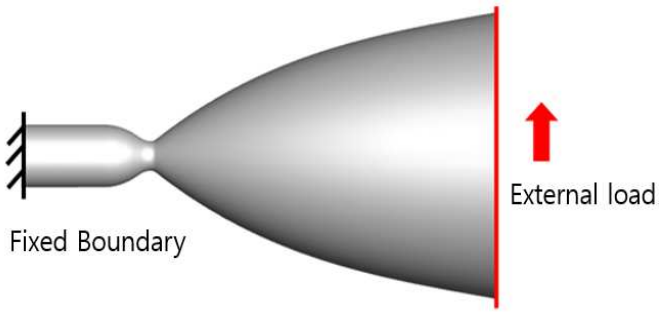


FIGURE 11. FETI-local analysis under tip loads with fixed boundary condition

NASTRAN is used to validate the proposed FETI-local static analysis under tip loads with a fixed boundary condition. The parameters for the FETI-local static analysis are presented in Table 5.

TABLE 5. Parameters for the FETI-local static analysis

Classification	**Value**
Young's modulus [GPa]	200
Thickness [m]	0.002
Poisson's ratio	0.3
Total tip loads [N]	1,000,000
Boundary condition	All fixed
Total degrees of freedom	120,300

Due to the limitation in the number of color contours available for both the present analysis and NASTRAN, Fig. 12 does not provide the exact displacement contours. However, the result still shows a good correlated displacement in each direction, when compared with those obtained by NASTRAN, within less than a 2% difference, as listed in Table 6.

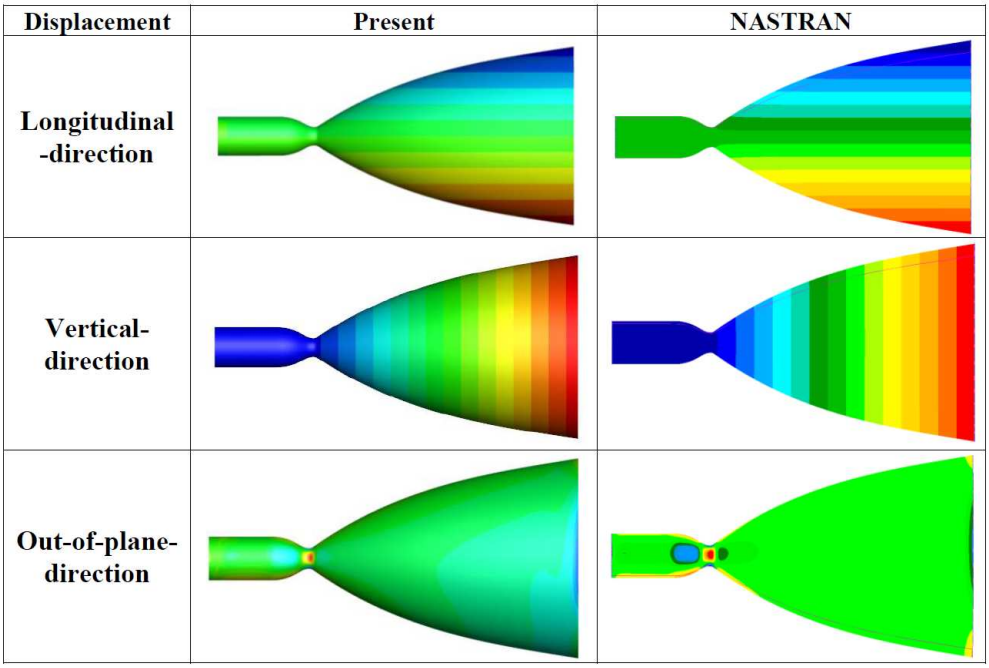


FIGURE 12. Displacement comparison against NASTRAN

TABLE 6. Tip displacement comparison against NASTRAN

Displacement	Present	NASTRAN	Discrepancy (%)
Longitudinal direction [m]	1.12×10^{-3}	1.14×10^{-3}	1.7
Vertical direction [m]	3.15×10^{-2}	3.2×10^{-2}	1.6
Out-of-plane direction [m]	-8.63×10^{-5}	-8.78×10^{-5}	1.7

The FETI-local static analysis is implemented in a parallel computing hardware. To conduct an evaluation of the FETI-local analysis, the computational time is compared with those of various CPUs. With this procedure, the number of degrees of freedom is kept to a total of 120,300, whereas the number of CPUs increases from 4 to 40.

Figures 13 and 14 show the trend of the computational time and memory usage while increasing the number of CPUs used from 4 to 40. The computational time and memory increase

when more than 40 CPUs are used. This is due to the increased degrees of freedom, as defined in Eq. (2.16). As the number of CPUs increases, the degrees of freedom for the localized Lagrange multipliers will also increase. Therefore, the number of CPUs should be carefully chosen to be as small as possible for efficient parallel computing. It shows a 94% computational time and memory usage reduction by using 20 CPUs, when compared with those obtained by using 4 CPUs. Therefore, the present FETI-local method is efficient for the reduction of computational time and its memory usage in parallel environments.

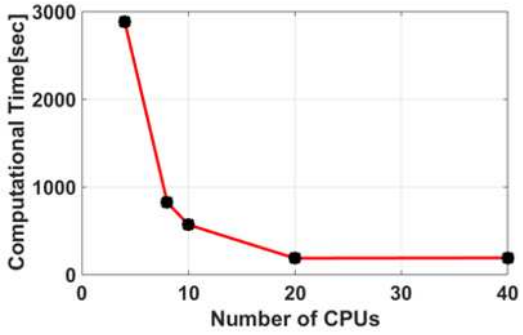


FIGURE 13. Scalability trend

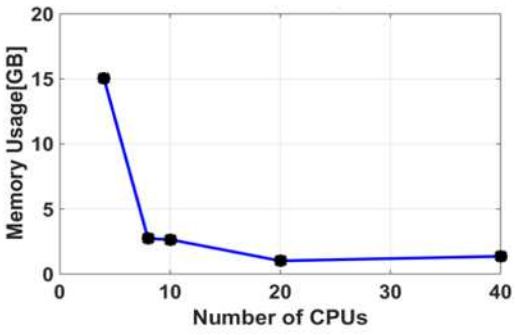


FIGURE 14. Memory usage trend

TABLE 7. Trend of computaional time and memory usage

CPUs	**4**	**8**	**10**	**20**	**40**
Computational Time [sec]	2932.75	824.42	569.49	188.42	191.99
Memory Usage [GB]	15.02	2.72	2.65	1.01	1.36

3.3. **FETI-local dynamic analysis of a cylinder.** Regarding the isotropic cylinder configuration, as shown in Fig. 5, the three-dimensional FETI-local dynamic analysis is conducted. Table 8 summarizes details of the present multi-core environment specifications. Its improved

TABLE 8. Specifications of the present multi-core environments

Processor	Intel® Xeon® E5-2420
Number of nodes	7
Number of CPU	108
Memory	500 GB
OS	CentOS 6.5
Memory transfer per second	56GB/s(InfiniBand)

TABLE 9. Parameters for the FETI-local dynamic analysis

Classification	**Value**
Young's modulus [GPa]	210
Thickness [m]	0.2
Poisson's ratio	0.3
Density	7850
Total tip loads [N]	9.6×10^5
Total element	18,432
Total nodes	9,240
Total degrees of freedom	55,440
Time step [sec]	0.01

memory transfer capability is used in the present dynamic analysis. Newmark-β method is employed. The key parameters of the dynamic analysis are summarized in Table. 9. Figure 15 is the comparison of the present result with ANSYS, for the vertical displacement at the cylinder tip. The present result is good agreement with that obtained by ANSYS, within a difference as small as 0.74% at the peak-to-peak.

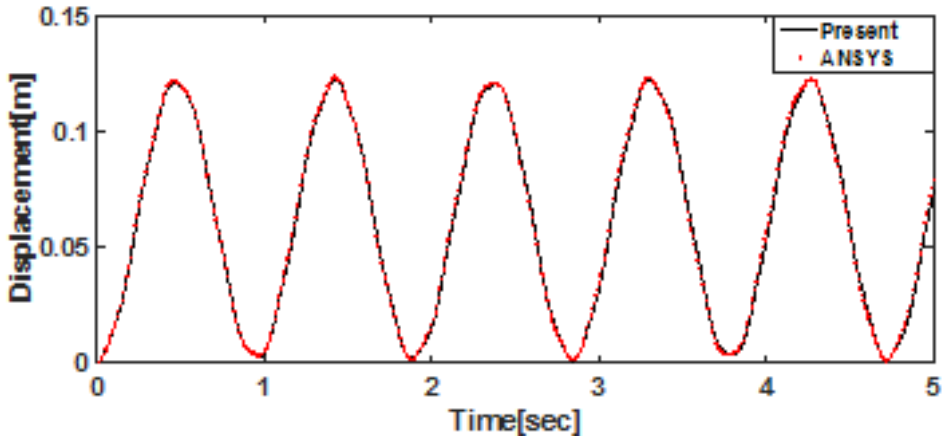


FIGURE 15. Comparison of the vertical displacement against ANSYS

Figures 16 and 17 show the trend of the computational time and memory usage while increasing the number of CPUs used from 4 to 48.

Table 10 summarizes the trend of computational time and memory usage. The computational time and memory increase further when more than 48 CPUS are used. This is due to the increased degrees of freedom for the localized Lagrange multipliers. Additionally for the computation time in Step, it is affected by the the size of the Boolean matrices, i.e., $\underline{\underline{K}}_{12}^{(i)}$,

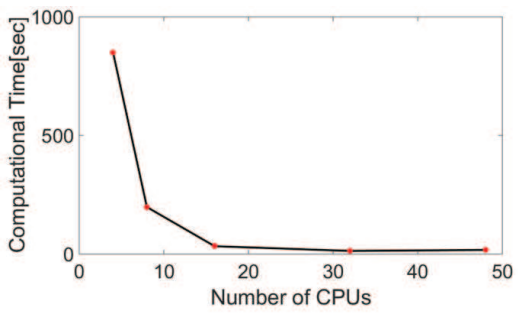


FIGURE 16. Scalability trend

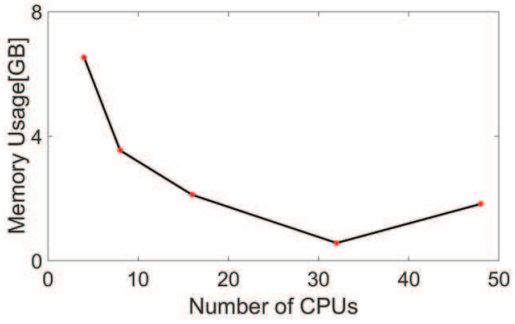


FIGURE 17. Memory usage trend

$\underline{\underline{K}}_{21}^{(i)}$ and $\underline{\underline{K}}_{22}^{(i)}$ which will increase in proportion to the number of the local Lagrange multipliers in Eq. (2.14). It shows a 98.5% computational time and memory usage reduction when using 32 CPUs, when compared with those obtained by using 4 CPUs. It is expected to obtain better efficiency in the dynamic analysis with a smaller size of the time step. Therefore, the present FETI-local method will be efficient in the reduction of computational time and its memory usage for the three-dimensional static analysis.

TABLE 10. Trend of computational time and memory usage

CPUs	**4**	**8**	**16**	**32**	**48**
Computational Time [sec]	850.383	197.9	32.9	12.978	17.15
Memory Usage [GB]	6.538	3.5463	2.11698	0.5713	1.82673

4. CONCLUSIONS

This paper describes the development of three-dimensional structural analysis with FETI-local method. The FETI-local method, is improved decomposition method from original FETI. The original FETI method consisted of direct solution approach for each of the sub-domain and iterative solvers for the interface problems. On the other hand, the FETI-local method is capable to obtain the solutions of each of the sub-domain and interface problem by applying the penalty term. The penalty term is added to the constraint problem and yields the ideal precondition. The solution procedure is introduced by three steps. At first step, decomposed stiffness matrices and load vectors are introduced from each CPU. Those terms are assembled into a root CPU, and localized Lagrange multipliers are calculated in the second step. Finally, displacement of each sub-domain can be obtained by localized Lagrange multipliers, its stiffness matrices and load vectors in the third step. A sparse direct solver library is used in every single CPU to handle the sparse stiffness matrices in the first and third step.

The three-dimensional FETI-local static structural analysis is implemented. The present result is validated in the different number of the sub-domains. The computational time and

memory usage are compared with various CPUs. The present result shows the decreasing trend of the computational time and memory usage when increasing the number of CPUs used from 4 to 32. It shows a 96.6% computational time and memory usage reduction by using 32 CPUs, when compared with those obtained by using 4 CPUs. It is confirmed the number of CPUs should be carefully chosen for efficient parallel computing.

The three-dimensional FETI-local dynamic analysis is conducted. The Newmark-β method is employed to extend the dynamic analysis. The present result is compared with the result obtained using ANSYS. The present result shows the decreasing trend of the computational time and memory usage when increasing the number of CPUs used from 4 to 32. It shows a 98.5% computational time and memory usage reduction by using 32 CPUs, when compared with those obtained by using 4 CPUs. It is expected to improve the computational efficiency by combining the advantages of the original FETI method, i.e., FETI-mixed using the mixed local-global Lagrange multiplier.

ACKNOWLEDGMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT and future Planning(2017R1A2B4004105) and supported by the EDISON Program through the National Research Foundation of Korea funded by the Ministry of Science, Information and Communication Technologies (ICT), and Future Planning (no. 2014M3C1A6038842).

REFERENCES

[1] Kwak, J. Y., Chun, T. Y., Shin, S. J., *Computational Mechanics*, Domain Decomposition Approach to Flexible Multibody Dynamics Simulation, Vol. 53, No. 1, pp. 147–158

[2] Farhat, C., and Roux, F. X., *International Journal for Numerical Methods in Engineering*, A Method of Finite Element Tearing and Interconnecting and its Parallel Solution Algorithm, Vol. 32, 1991, pp. 1205–1227.

[3] C. Farhat, M. Lesoinne, P. Letallec, K. Pierson, D. Rixen, *Int. J. Numer. Meth. Engng*, FETI-DP: a dual-primal unified FETI Method- Part I: A Faster Alternative to the Two-level FETI Method, 50, pp. 1523-1554, 2001.

[4] Park, K. C., Fellipa, C. A., and Gumaste, U. A., *Computational Mechanics*, A Localized Version of the Method of Lagrange Multipliers and its Applications, Vol. 24, Issue 6, 2000, pp. 476–490.

[5] Bauchau, O. A, Epple, A., and Bottasso, C. L., *Journal of Computational and Nonlinear Dynamics*, Scaling of Constraints and Augmented Lagrangian Formulations in Multibody Dynamics Simulations, Vol. 4, 2009.

[6] Bauchau, O. A., *Journal of the Franklin Institute*, Parallel Computation Approaches for Flexible Multibody Dynamics Simulations, Vol. 347, No. 1, 2010, pp. 53–68

[7] Kwak, J. Y., Chun, T. Y., and Shin, S. J., Computational Approaches for Large Scale Structural Analysis using Domain Decomposition Technique, 52^{nd} AIAA/ASME/ASCE/AHS Structures, Structural Dynamics and Materials Conference 19^{th} 4-7 April 2011, Denver, Colorado

[8] Vondrak, V., Dostal, Z., Dobias, J., and Ptak, S., *Lecture Notes in Computational Science and Engineering*, Domain Decomposition Methods in Science and Engineering XVI, pp771-778

[9] Kwak, J. Y., Chun, T. Y., Cho H. S., and Shin, S. J., *Journal of the Korean Society for Industrial and Applied Mathmatics*, Advanced Domain Decomposition Method by Local and Mixed Lagrange Multipliers, 18(2.1), 17-26, 2011

[10] Hashamdar, H., Ibrahim, Z., Jameel, M., *International Journal of the Physical Sciences*, Finite Element Analysis of Nonlinear Structures with Newmark Method, Vol. 6(2.6), 1395-1403, 2011

[11] Felippa, C. A.,*Computer Methods in Applied Mechanics and Engineering*, A Study of Optimal Membrane Triangles with Drilling Freedoms, Vol. 192, No. 16, 2003, pp. 2125–2168.
[12] Batoz J. L., Bathe K. J., and Ho L. W., *International Journal for Numerical Methods in Engineering*, A Study of Three-node Triangular Plate Bending Elements, Vol. 15, 1980, pp. 1771–1812.
[13] Khosravi, P., Ganesan, R., and Sedaghati, R., *International Journal for Numerical Methods in Engineering*, Corotational Non-linear Analysis of Thin Plate and Shells using a New Shell Element, Vol. 69, 2007, pp. 859–885.
[14] Kwak, J. Y., Cho, H., Chun, T. Y., Shin, S. J., *International Journal of Aeronautical and Space Sciences*, Domain Decomposition Approach Applied for Two- and Three-dimensional Problems via Direct Solution Methodology, 16(2.2), 177-189, 2015
[15] Allman, J., *International Journal for Numerical Methods in Engineering*, Evaluation of the Constant Strain Triangle with Drilling Rotations, Vol. 26, 1988, pp. 2645–2655.