

## 3D Nonlinear Parallel FEM for Analyzing Dynamic Response of the Large-Scale Saturated Soil Layers-Civil Structures Interaction Problem (Part I: Formulation and its Numerical Solution)

**Jafril Tanjung**

Civil Engineering Department, Engineering Faculty, Andalas University, Padang  
Email: jafriltanjung@ft.unand.ac.id, jafril\_tanjung@yahoo.com

### Abstract

*In this analytical study a 3D nonlinear parallel FEM algorithm was developed to overcome the physical limitation of a single processor computer when analysing the dynamic response of interaction between a large-scale saturated soil layers and civil structures. The parallel algorithm was derived based on the Domain Decomposition Method (DDM) such that a boundary value problem can be converted into an interface problem. The DDM was used to partite a whole of analytical domain into several non-overlapping subdomains. One feature of the current algorithm is most parts of the computation processes are performed in subdomain basis without interprocessor communication. The interprocessor communication is only take place when solving the interface problem. An iterative solver of the Conjugate Gradient (CG) algorithm incorporating with the interprocessor communication based on hypercube networking was used to solve it interface problem.*

**Keywords:** 3D nonlinear FEM, parallel computation, domain decomposition method, saturated soil layers-civil structure interaction problem.

### Abstrak

*Dalam studi analitik ini, algoritma paralel elemen hingga non-linier dikembangkan untuk mengatasi kelemahan dari kemampuan komputer berprosesor tunggal dalam menganalisis tanggap dinamik pada masalah interaksi antara lapisan tanah jenuh dan struktur sipil dalam skala besar. Algoritma paralel dijabarkan berdasarkan metode dekomposisi domain untuk mengubah masalah nilai batas ke masalah interfes. Metode dekomposisi domain digunakan untuk membagi keseluruhan domain yang dinalisis menjadi beberapa subdomain yang tidak saling overlap. Keunggulan dari algoritma yang dikembangkan ini adalah sebagian besar proses perhitungan dilakukan dalam basis subdomain tanpa adanya komunikasi antar prosesor. Komunikasi antar prosesor hanya dilakukan ketika menyelesaikan masalah interfes. Metode iteratif Conjugate Gradient, yang digabungkan dengan komunikasi antar prosesor berbasisan jejaring hypercube, digunakan untuk menyelesaikan masalah interfes tersebut.*

**Kata-kata Kunci:** Metode elemen hingga non-linier 3-dimensi, paralel komputasi, metode dekomposisi domain, masalah interaksi antara lapisan tanah jenuh dan struktur sipil.

## 1. Introduction

The quay walls structures constructed as the waterfront structure in the port area often damaged during great earthquake. The recent example is most of quay walls constructed in the port of Kobe were damaged by the Hyogo-ken Nanbu earthquake on January 17<sup>th</sup>, 1995 (Baker and Smith, 1987). The damaged of these civil structures are indicating the complexity of the interaction problem between saturated soil layers and the civil structures. The source of complexity mostly comes from the dynamic response of the saturated soil layers where the soil grains and the pore water interacts each other.

To understand the dynamic behaviour of the water-front structures which retain the saturated soil layers, the dynamic interaction between saturated soil layers and civil structures should be analyzed more exactly. There exist two problems for analysis the dynamic interaction problem mentioned above. The first problem is how to idealize the saturated soil layer. It is required to assume the saturated soil layer as a composite of the soil grains and the pore water. Analysis of the saturated soil layers considering the coupled system soil grains and pore water gives more realistic results compared with homogeneous equivalent solid assumption. It is also necessary to use an appropriate material constitutive model for the soil grains, which able to represent the nonlinear stress-strain relation-

ship and the loading-unloading hysteresis loop. To solve the first problem, a simplified bounding surface plasticity model was introduced referring the previous studies by Dafalias (1986) and Crouch and Wolf (1994). In this model the change of plastic strain during cyclic loading is evaluated according to relative position of the current stress point on the loading surface inside the bounding surface. The model is applicable for both of clays and sands by unifying the parameters in the single frame of a constitutive model. The number of parameters for this model was reduced to specify the model more easily.

The second problem is how to reduce heavy loads on the computation process. Long computing time and large computer memory are needed to solve three-dimensional coupling problem applying iterative method for nonlinear analysis and time history response analysis for a recorded earthquake motion. Analyses using conventional single processor very often exceed the available computer capacity. With advent the computer technology within last decades, analyzing a large-scale problem using several processor concurrently becomes possible. This type of works is called parallel computation.

There are several proposed parallel algorithms in the field of the finite element analysis as well documented in the literatures such as Tallec (1997), Papadarakakis (1997), Farhat and Raux, (1991). However, most of them were used for analyzing the homogenous or equivalent homogenous solid materials. In the previous studies, the spatial discretization for the finite element approximation was derived based on the variational method, which is not valid for the analysis of the fluid material. Therefore, in this study, the Weighted Residual Method (WRM) was applied to define the spatial discretization of the finite element and uses a shape function as the weight function as have been recommended by the Galerkin Method. WRM is more suitable for solid as well as for fluid such as a saturated soil layer.

The parallel algorithm developed herein is based the Domain Decomposition Method (DDM) (Tallec, 1997, Papadarakakis, 1997, Farhat and Raux, 1991) for separating a whole of interest domain into several non-overlapping subdomains. A traction force was used to enforce the continuity condition on the subdomain interfaces, i.e. the summation of the unknowns on the interfaces are equal to zero. As the result, a boundary value problem then can be converted into an interface problem.

To achieve an efficient parallel algorithm, two following conditions should be considered, i.e. load works balancing and interprocessor communication. The works in each processor should be balanced as possible and the interprocessor communication should

be as less as possible. The effectiveness of the interprocessor communication is related to the scheduling and the communication algorithms (Kawamura and Tanjung, 2000). In order to reach these conditions, the proposed parallel algorithm was design to be mostly work in subdomain basis without interprocessor communication. For this purpose, the algorithm was implemented as a Single Program Multiple Data (SPMD) programming model. The interprocessor is only required when solving the interface problem.

## 2. Governing Equation of Motion

The governing equations of motion were derived by assuming the saturated soils compose of the soil grains and the pore water. The density of the saturated soil,  $\rho$  is expressed by the density of the soil grains  $\rho_g$  and the fluid density  $\rho_f$ , as schematically shown in Figure 1.

$$\rho = (1 - v)\rho_g + v\rho_f \quad (1)$$

Assuming infinitesimal displacement, the equations of motions for saturated porous material are given as follows (Ghaboussi, 1973, Zienkiewicz, 1984).

$$\sigma_{ij,i} - \rho \ddot{u}_i - \rho_f \ddot{w}_i = 0 \quad (2)$$

$$p_{,i} - \rho_f \left( \ddot{u}_i + \frac{\ddot{w}_i}{f} \right) - \frac{\rho_f g}{k} \dot{w}_i = 0 \quad (3)$$

where  $\sigma_{ij}$  are the components of total stress of the saturated soil,  $p$  is the pore water pressure;  $u_i$  and  $w_i$  are the components of displacement of the soil grain and the relative displacement of the pore water to  $u_i$ , respectively. The superposed dot implies time derivative;  $(\cdot)_{,i}$  denotes the derivative with respect to coordinates  $x_i$ . For  $\sigma_{ij}$  and  $p$  tensile stresses are defined positive.  $k$  is the permeability and  $g$  is the gravitational acceleration.

where  $v$  is the porosity.

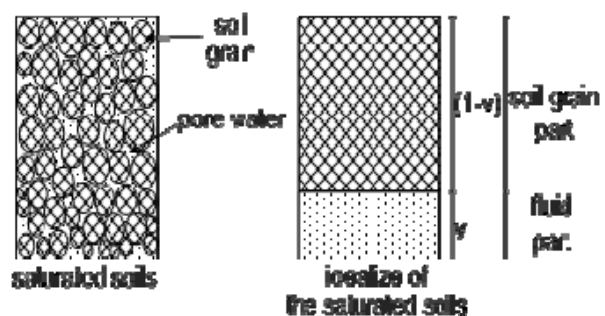


Figure 1. Idealization of saturated soils  
(Source: Tanjung, 2002)

### 3. Constitutive Model

A constitutive relation for the saturated soil layer was derived for the total stress, i.e. as a summation of the effective stress and the pore water pressure. For the pore water, the constitutive relation was obtained by applying the mass conservation law to the flow of the pore water in saturated soil layer. Their increment form is given as follow (Dafalias, 1986).

$$\dot{\sigma}_{ij} = D_{ijkl} \dot{\epsilon}_{kl} + \alpha^2 \delta_{ij} K_{gf} \delta_{kl} \dot{\epsilon}_{kl} + \alpha K_{gf} \delta_{ij} \dot{\zeta} \quad (4)$$

$$\dot{p} = K_{gf} (\alpha \dot{\epsilon}_{ii} + \dot{\zeta}) \quad (5)$$

$$\frac{1}{K_{gf}} = \frac{1}{K_g} + \frac{1}{K_f} \quad (6)$$

$$\alpha = 1 - \frac{\delta_{ij} D_{ijkl} \delta_{kl}}{3K_g} \quad (7)$$

$\dot{\sigma}_{ij}$  and  $\dot{p}$  are the total stress and pore water pressure

increment, respectively.  $K_g$ ,  $K_f$  and  $K_{gf}$  are the bulk modulus for the soil grain, pore water and their coupling, respectively.  $\alpha$  is a measure of compressibility of the soil grains representing the contact areas of the soil grains to pore water.  $D_{ijkl}$  represents the stress-strain relation of soil grain.  $d_{ij}$  is the component of the Kronecker's delta.

$\dot{\epsilon}_{ij}$  and  $\dot{\zeta}$  are the strain of soil grain and volumetric

strain of the pore water increments, respectively. By using the small deformation theory, their values are defined as follow.

$$\dot{\epsilon}_{ij} = \frac{1}{2} (u_{i,j} + u_{j,i}) \quad (8)$$

$$\dot{\zeta} = w_{i,i} \quad (9)$$

Elastoplastic stress-strain relation  $D_{ijkl}^{ep}$  for the general

type of yield function may be expressed as following equation (Dafalias, 1986).

$$D_{ijkl}^{ep} = D_{ijkl}^e - u(L) \frac{D_{ijmn}^e P_{mn} Q_{pq} D_{pqkl}^e}{H + Q_{ab} D_{abcd}^e P_{cd}} \quad (10)$$

in which

$$L = \frac{Q_{ij} D_{ijkl}^e \dot{\epsilon}_{kl}}{H + Q_{ab} D_{abcd}^e P_{cd}} \quad (11)$$

$$P_{ij} = \frac{\partial g_p}{\partial \sigma'_{ij}} \quad (12)$$

$$Q_{ij} = \frac{\partial f}{\partial \sigma'_{ij}} \quad (13)$$

$D_{ijkl}^e$  represents the elastic stress-strain relation,  $H$  is the plastic hardening modulus,  $g_p$  and  $f$  are the plastic potential and the yield functions.  $u(L)$  is the heavy-step function,  $u(L)=1$  for  $L \geq 0$  and  $u(L)=0$  for  $L < 0$ . In the bounding surface plasticity model, these relations are applied not only for the bounding surface but also for the loading surface inside the bounding surface. To specify the relations in **Equation 10** in detail, the definition of the elastic stress-strain relation  $D_{ijkl}^e$ , the failure surface  $F_f$ , the bounding surface  $F_b$ , the loading surface  $f_L$  and the plastic strain potential  $g_p$  should be determined. The form of the functions mentioned above is explained in the followings.

#### 3.1 Elastic stress-strain relation

The elastic stress-strain relation  $D_{ijkl}^e$  is expressed as

$$D_{ijkl}^e = 2G \delta_{ik} \delta_{jl} + \left( K_g - \frac{2}{3} G \right) \delta_{ij} \delta_{kl} \quad (14)$$

The notation  $K_g$  and  $G$  denote bulk and shear modulus of the soil grains, respectively. They are defined based on the critical state of soil mechanics concept as written in the following expressions for a given constant Poisson's ratio  $\nu$ .

$$\frac{K_g}{p_a} = \frac{1 + e_{in}}{\kappa} \left( \frac{I}{3p_a} \right)^{0.5} \quad (15)$$

$$G = \frac{3(1-2\nu)}{2(1+\nu)} K_g \quad (16)$$

Notation  $e_{in}$  indicates the initial void ratio,  $\kappa$  is the slope of the swelling process in the consolidation curve,  $p_a$  is the atmosphere pressure,  $I$  is the first invariant of stress.

#### 3.2 Failure surface

An expression for failure surface  $F_f$  is defined as follows.

$$F_f = -\zeta_{cr} I + J = 0 \quad (17)$$

$J$  is the second invariant of deviatoric stress;  $V_{cr}$  is the slope of the critical state line in the meridional section shown in **Figure 3**. In the effective stress space, the critical state is represented by an elliptic cone function  $\rho_c$  as proposed by Crouch and Wolf (1994) as follow.

$$\zeta_{cr} = \rho_c \frac{2 \sin \phi_{cr}}{\sqrt{3} (3 - \sin \phi_{cr})} \quad (18)$$

The shape of the cone is shown in **Figure 2**. The shape of the cone is controlled by the Lode angle  $\theta$  and a material parameter defining the compression critical friction angle  $\phi_{cr}$ .

### 3.3 Bounding surface

A single elliptic surface shown in **Figure 3** was used as the bounding surface in this study. The parameter  $R$  controls the aspect ratio of the ellipse. The form of the bounding surface  $F_b$  is given in following equation (Tanjung, 2002).

$$F_b = \bar{J}^2 - \frac{(\zeta_{cr} I_0)^2}{R^2} \left( 1 - \frac{1}{(R-1)^2} \left( \frac{R\bar{I}}{I_0} - 1 \right)^2 \right) = 0 \quad (19)$$

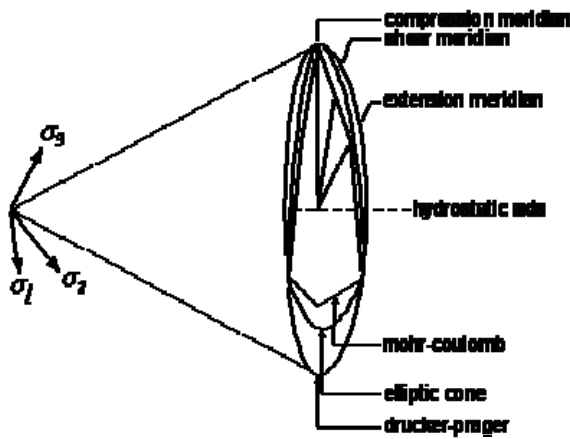


Figure 2. Elliptic cone shape of failure surface  
(Source: 2.Crouch, at.al., 1994)

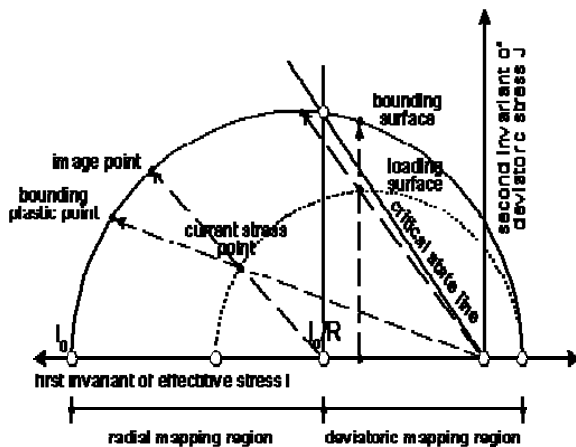


Figure 3. Single elliptic of bounding surface model  
(Source: Tanjung, 2002)

The size of bounding is defined by a maximum isotropic compression stress  $I_0$ . The bounding surface moves during plastic flow and always envelopes the current effective stress state. The evolution of the bounding surface depends on the volumetric plastic strain increment  $\varepsilon_v^p$  as expressed in **Equation (20)**. The bounding surface may expand or shrink depends upon whether the plastic volumetric  $\varepsilon_v^p$  increases or decreases.

$$\frac{\dot{I}_0}{p_a} = \frac{1 + e_{in}}{\lambda_c - \kappa} \left( \frac{I_0}{3p_a} \right) \dot{\varepsilon}_v^p \quad (20)$$

$\lambda_c$  is a slope of isotropic consolidation line.

### 3.4 Loading surface

A loading surface is required to define the loading direction and to evaluate the plastic condition on the current effective stress point. The shape of the loading surface was taken similar with the bounding surface form. The plastic condition of the loading surface, which is located inside the bounding surface, is related to the plastic condition on the bounding surface.

$(\bar{I}, \bar{J})$  on the bounding surface. In order to capture

As shown in **Figure 3**, this relation is taken by mapping the current stress point  $(I, J)$  to the image point

the wide range of the soil responses, the following combination of radial and deviatoric mapping rules have been applied (Tanjung, 2002).

$$\bar{I} = \left( \frac{I_0}{R} \right) + \beta \left( I - \frac{I_0}{R} \right); \quad \bar{J} = \beta J; \quad \text{for } I < I_0/R$$

(radial mapping) (21)

$$\bar{I} = I; \quad \bar{J} = \beta J \quad \text{for } I \geq I_0/R \quad (\text{deviatoric mapping}) \quad (22)$$

A constant  $\beta$  called a similarity ratio. The equations which define the loading surfaces in the radial and deviatoric mapping regions can be obtained by substituting **Equations 21** and **22** into the bounding surface equation, **Equation 19**, respectively.

$$f_L = \beta^2 J^2 - \frac{(\zeta_{cr} I_0)^2}{R^2} \left( 1 - \frac{\beta^2}{(R-1)^2} \left( \frac{R\bar{I}}{I_0} - 1 \right)^2 \right) = 0 \quad (23)$$

$$f_L = \beta^2 J^2 - \frac{(\zeta_{cr} I_0)^2}{R^2} \left( 1 - \frac{1}{(R-1)^2} \left( \frac{R\bar{I}}{I_0} - 1 \right)^2 \right) = 0 \quad (24)$$

### 3.5 Potential surface

The potential surface  $g_p$  is also assumed as the similar form with the bounding surface.

$$g_p = J^2 - \frac{(\zeta_{cr} I_{0g})^2}{R^2} \left( 1 - \frac{1}{(R-1)^2} \left( \frac{RI}{I_{0g}} - 1 \right)^2 \right) = 0 \quad (25)$$

in which subscript  $g$  denotes the parameters related to the potential surface  $g_p$ . All parameters except  $I_{0g}$  are same with those of the bounding surface. A parameter  $I_{0g}$  is scaled such that the current stress point lies on the potential surface. Although the bounding, the loading and the potential surfaces are the similar elliptic form, these surfaces are not geometrically identical and the responses of the soil are always obtained in the non-associated flow rule.

### 3.6 Plastic hardening modulus

The consistency condition for the yield surface is required to maintain the effective stress point always lies within or on the yield surface. For the bounding surface plasticity model, the consistency condition is applied to the bounding surface function  $F_b$ . The plastic hardening modulus  $H$  on the loading surface is defined referring the value of the plastic hardening modulus  $\bar{H}$  on the bounding surface.

$$H = \bar{H} + H_f \quad (26)$$

in which  $\bar{H}$  is evaluated by the following equation.

$$\bar{H} = -\frac{1+e_{in}}{\lambda-\kappa} I_0 \frac{\partial F}{\partial I_0} P_{ii} \quad (27)$$

A term  $\partial F / \partial I_0$  is considered for the bounding plastic point on the bounding surface. Its point is determined by projecting the current stress point from the origin of the stress space onto the bounding surface using the following mapping rule.

$$\bar{I} = \beta I; \bar{J} = \beta J \quad (28)$$

As was previously mentioned, the feature of the bounding surface plasticity model is that the plastic strain is allowed to occur not only on the bounding surface, but also inside the bounding surface. This plastic strain inside the bounding surface can be considered as the relative plastic strain to the plastic strain on the bounding surface. To take into account the plastic strain inside the bounding surface, Crouch and Wolf (1994) proposed an additional plastic hardening modulus  $H_f$  as follow.

$$H_f = \frac{1+e_{in}}{\lambda-\kappa} \frac{h}{\langle \beta / (\beta - 1) - s_e \rangle} \left( \frac{I_0}{3p_a} \right)^{1.5} \quad (29)$$

A constant  $h$  is a material parameter represents the hardening behaviour of the soil grains and parameter  $s_e$  defines the elastic nucleus where an only elastic strain is developed.

## 4. Nonlinear Dynamic Parallel

### 4.1 Applied DDM in FEM Formulation

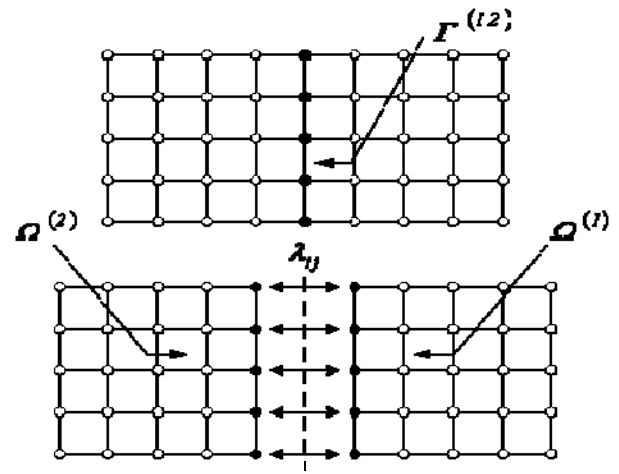
To clarify usage of a Domain Decomposition Method (DDM) in the nonlinear FEM formulation, consider a case of a domain  $\Omega$  separated into two totally non-overlapping subdomains  $\Omega^{(1)}$  and  $\Omega^{(2)}$  shown in **Figure 4**. Extrapolation to number of  $N_s$  subdomains can be done in a straightforward. The  $\Gamma^{(12)}$  is the interface between subdomain  $\Omega^{(1)}$  and subdomain  $\Omega^{(2)}$ . Under constraint of the displacements field on the interface the additional boundary condition as written in **Equation 30** is required with the continuity condition given in **Equation 31** (Kawamura and Tanjung, 2000, Kawamura and Tanjung, 2001).

$$\begin{Bmatrix} \lambda_{ij}^u n_j \\ \lambda_{ij}^w n_j \end{Bmatrix}^{(1)} + \begin{Bmatrix} \lambda_{ij}^u n_j \\ \lambda_{ij}^w n_j \end{Bmatrix}^{(2)} = 0 \quad \text{on } \Gamma^{(12)} \quad (30)$$

$$\begin{Bmatrix} u_i n_j \\ w_i n_j \end{Bmatrix}^{(1)} = \begin{Bmatrix} u_i n_j \\ w_i n_j \end{Bmatrix}^{(2)} \quad \text{on } \Gamma^{(12)} \quad (31)$$

$\lambda_{ij}$  denotes a traction tensor and  $n_j$  is an outer normal vector.

A solution of the above problem is obtained by applying the Weighted Residual Method (WRM) (Zienkiewicz and Taylor, 1991, Hughes, 1996) to the governing equations of motion (2) and (3), the continuity condition on the interface (31) and the boundary condition (30).



**Figure 4. Decompose into two subdomains**  
(Source: Tanjung, 2002)

$$\begin{aligned}
 & \sum_{s=1}^2 \left[ \int_{\Omega^{(s)}} \{ \delta u_i \quad \delta w_i \}^{(s)} \left\{ \begin{matrix} \sigma_{ij,j} \\ p_i \end{matrix} \right\}^{(s)} d\Omega^{(s)} \right. \\
 & - \int_{\Omega^{(s)}} \{ \delta u_i \quad \delta w_i \}^{(s)} \left\{ \begin{bmatrix} \rho & \rho_f \\ \rho_f & \frac{1}{f} \rho_f \end{bmatrix} \left\{ \begin{matrix} \ddot{u}_i \\ \ddot{w}_i \end{matrix} \right\} \right\}^{(s)} d\Omega^{(s)} \\
 & - \int_{\Omega^{(s)}} \{ \delta u_i \quad \delta w_i \}^{(s)} \left\{ \begin{bmatrix} 0 & 0 \\ 0 & \frac{\rho_f g}{k} \end{bmatrix} \left\{ \begin{matrix} \dot{u}_i \\ \dot{w}_i \end{matrix} \right\} \right\}^{(s)} d\Omega^{(s)} \left. \right] \\
 & + \int_{\Gamma^{(12)}} \{ \delta u_i \quad \delta w_i \}^{(12)} \left\{ \begin{bmatrix} \lambda_{ij}^u n_j \\ \lambda_{ij}^w n_j \end{bmatrix}^{(1)} + \begin{bmatrix} \lambda_{ij}^u n_j \\ \lambda_{ij}^w n_j \end{bmatrix}^{(2)} \right\} d\Gamma^{(12)} = 0
 \end{aligned} \quad (32)$$

$$\int_{\Gamma^{(12)}} \{ \delta u_i^u n_j \quad \delta u_i^w n_j \}^{(12)} \left\{ \begin{bmatrix} u_i n_j \\ w_i n_j \end{bmatrix}^{(1)} - \begin{bmatrix} u_i n_j \\ w_i n_j \end{bmatrix}^{(2)} \right\} d\Gamma^{(12)} = 0 \quad (33)$$

Superscript (s) denotes the subdomain number,  $\delta u_i$  and  $\delta w_i$  are the arbitrary functions which represent the weight for approximating the values of the displacements  $u_i$  and  $w_i$ . It is possible to reduce an order of differentiation in the first term of left hand side of **Equation 32** via integration by parts of Green's theorem. The spatial discretization for the finite element is then obtained by introducing the weight function for **Equation 32** and **33**. The Galerkin method (Zienkiewicz and Taylor, 1991, Hughes, 1996) was applied in this study; i.e. uses an original shape function as a weight function. The final equation of motion for the finite element analysis is obtained as written in **Equation 34**.

$$\sum_{s=1}^2 \left[ \mathbf{M}^{(s)} \ddot{\mathbf{d}}^{(s)} + \mathbf{D}^{(s)} \dot{\mathbf{d}}^{(s)} + \mathbf{K}^{(s)} \mathbf{d}^{(s)} \right] = \sum_{s=1}^2 \left[ \mathbf{F}^{(s)} - \mathbf{B}^{(s)T} \boldsymbol{\lambda} \right] \quad (34)$$

with respect to continuity condition,

$$\sum_{s=1}^2 \mathbf{B}^{(s)} \mathbf{d}^{(s)} = \mathbf{0} \quad (35)$$

in which

$$\mathbf{d}^{(s)} = \begin{Bmatrix} \mathbf{d}_u^{(s)} \\ \mathbf{d}_w^{(s)} \end{Bmatrix}; \mathbf{M}^{(s)} = \begin{bmatrix} \mathbf{M}_u^{(s)} & \mathbf{M}_c^{(s)} \\ \mathbf{M}_c^{(s)T} & \mathbf{M}_w^{(s)} \end{bmatrix}; \mathbf{D}^{(s)} =$$

$$\begin{bmatrix} \mathbf{D}_u^{(s)} & 0 \\ 0 & \mathbf{D}_w^{(s)} \end{bmatrix}; \mathbf{K}^{(s)} = \begin{bmatrix} \mathbf{K}_{Tu}^{(s)} & \mathbf{K}_c^{(s)} \\ \mathbf{K}_c^{(s)T} & \mathbf{K}_w^{(s)} \end{bmatrix}$$

$\mathbf{M}^{(s)}$ ,  $\mathbf{D}^{(s)}$  and  $\mathbf{K}^{(s)}$  are the mass, damping and stiffness matrices for each subdomain.  $\mathbf{K}_{Tu}^{(s)}$  is the tangent stiffness matrix for the soil grains part, which depends on the effective stress history. The detailed formulation of these matrices for an 8-node brick element was written in reference (Kawamura and Tanjung, 2000). The subscript  $u$  and  $w$  denotes the soil grains and the pore water, respectively, the subscript  $c$  designates the coupling condition. The vectors  $\mathbf{d}_u^{(s)}$  and  $\mathbf{d}_w^{(s)}$  are the relative displacement of the soil grains and the pore water to the ground motion, respectively. The vector  $\boldsymbol{\lambda}$  is composed of  $\lambda_{ij}$  which is traction forces on the interface nodes. The matrix  $\mathbf{B}^{(s)}$  is signed Boolean matrix, which localizes a subdomain quantity to the subdomain interface. If the interior degrees of freedom are numbered first and the interface ones are numbered last, the Boolean matrix  $\mathbf{B}^{(s)}$  takes a form,

$$\mathbf{B}^{(s)} = [\mathbf{0} \quad \mathbf{I}] \quad (36)$$

In above equation,  $\mathbf{0}$  and  $\mathbf{I}$  denote a null and identity matrices.

#### 4.2 Numerical solution of the equation of motion

Perhaps the most widely used the direct time integrating method for solving the equation of motion such as written in **Equation 34** is the Newmark's family (Newmark, 1959). In this study, the Hilber- $\alpha$  (Hughes, 1996) method has been used. In this study, the following approximation was used to integrate equation of motion for constant increment time step  $\Delta t$  following the Hilber- $\alpha$  method.

$$\mathbf{d}_n^{(s)} = \mathbf{d}_{n-1}^{(s)} + \Delta \mathbf{d}_n^{(s)} \quad (37)$$

$$\dot{\mathbf{d}}_n^{(s)} = \frac{\bar{\gamma}}{\bar{\beta} \Delta t} \Delta \mathbf{d}_n^{(s)} + \left( 1 - \frac{\bar{\gamma}}{\bar{\beta}} \right) \dot{\mathbf{d}}_{n-1}^{(s)} + \left( 1 - \frac{\bar{\gamma}}{2\bar{\beta}} \right) \Delta t \ddot{\mathbf{d}}_{n-1}^{(s)} \quad (38)$$

$$\ddot{\mathbf{d}}_n^{(s)} = \frac{1}{\bar{\beta} \Delta t^2} \Delta \mathbf{d}_n^{(s)} - \frac{1}{\bar{\beta} \Delta t} \dot{\mathbf{d}}_{n-1}^{(s)} + \left( 1 - \frac{1}{2\bar{\beta}} \right) \ddot{\mathbf{d}}_{n-1}^{(s)} \quad (39)$$

The parameters  $\bar{\alpha}$ ,  $\bar{\gamma}$  and  $\bar{\beta}$  are specified as follow.

$$\frac{1}{3} \leq \bar{\alpha} \leq 0; \bar{\gamma} = \frac{1-2\bar{\alpha}}{2}; \bar{\beta} = \frac{(1-\bar{\alpha})^2}{4} \quad (40)$$

At first, subscripts  $n$  and  $k$  in the equations of motion (34) and (35) are introduced for signing a time step and nonlinear iteration number.

Substituting **Equations 37 to 39** into **Equations 34**

$$\sum_{s=1}^{Ns} \left[ \hat{\mathbf{K}}_{T_n}^{(s)} \Delta \mathbf{d}_n^{(s)} \right] = \sum_{s=1}^{Ns} \left[ {}^{(k)}\Delta \mathbf{R}_n^{(s)} - \mathbf{B}^{(s)T} {}^{(k)}\Delta \boldsymbol{\lambda}_n \right] \quad (41)$$

and **35** lead to following expression. The **Equations 41 to 43** have been extended to  $N_s$  subdomains instead of 2 subdomains derived in foregoing

$$\sum_{s=1}^{Ns} \left[ \mathbf{B}^{(s)} \hat{\mathbf{K}}_{T_n}^{(s)-1} \mathbf{B}^{(s)T} \right] {}^{(k)}\Delta \boldsymbol{\lambda}_n = \sum_{s=1}^{Ns} \left[ \mathbf{B}^{(s)} \hat{\mathbf{K}}_{T_n}^{(s)-1} {}^{(k)}\Delta \mathbf{R}_n^{(s)} \right] \quad (42)$$

with respect to continuity condition

$$\sum_{s=1}^{Ns} \mathbf{B}^{(s)} {}^{(k)}\Delta \mathbf{d}_n^{(s)} = \mathbf{0} \quad \text{for } n = 1, 2, \dots, N_s \quad (43)$$

A boundary value problem written in **Equation 47** may be converted into following interface

$$\hat{\mathbf{K}}_{T_n}^{(s)} = \frac{1}{\beta \Delta t^2} \mathbf{M}^{(s)} + \frac{\bar{\gamma}}{\beta \Delta t} \mathbf{D}_{n-1}^{(s)} + \mathbf{K}_{T_{n-1}}^{(s)} \quad (44)$$

$${}^{(k)}\Delta \mathbf{R}_n^{(s)} = \hat{\mathbf{F}}_n^{(s)} - \sum_{j=1}^{k-1} \left( \hat{\mathbf{K}}_{T_n}^{(s)} {}^{(j)}\Delta \mathbf{d}_n^{(s)} \right) \quad \text{lem.} \quad (45)$$

$$\begin{aligned} \Delta \hat{\mathbf{F}}_n^{(s)} &= \left( \mathbf{F}_n^{(s)} - \mathbf{F}_{n-1}^{(s)} \right) + \left( \frac{1}{\beta \Delta t} \mathbf{M}^{(s)} + \frac{\bar{\gamma}}{\beta} \mathbf{D}_{n-1}^{(s)} \right) \dot{\mathbf{d}}_{n-1}^{(s)} \\ &+ \left( \frac{1}{2\beta} \mathbf{M}^{(s)} + \left( \frac{\bar{\gamma}}{2\beta} - 1 \right) \Delta t \mathbf{D}_{n-1}^{(s)} \right) \ddot{\mathbf{d}}_{n-1}^{(s)} \end{aligned} \quad (46)$$

Obviously, there are three unknowns which have to be solved in the equation of motion (41), i.e.  $\mathbf{K}_{T_n}^{(s)}$ ,  ${}^{(k)}\Delta \boldsymbol{\lambda}_n$

$${}^{(k)}\Delta \mathbf{d}_n^{(s)} = \hat{\mathbf{K}}_{T_n}^{(s)-1} \left( {}^{(k)}\Delta \mathbf{R}_n^{(s)} - \mathbf{B}^{(s)T} {}^{(k)}\Delta \boldsymbol{\lambda}_n \right) \quad (47)$$

and  ${}^{(k)}\Delta \mathbf{d}_n^{(s)}$ . According to the Modified Newton-Raphson Method (Press, 1995) applied for solving the nonlinear condition in this study; a current tangent stiffness matrix  $\mathbf{K}_{T_n}^{(s)}$  may

$$\frac{{}^{(k)}\Delta \mathbf{R}_n^{(s)T} \sum_{j=1}^k {}^{(j)}\Delta \mathbf{d}_n^{(s)}}{\Delta \hat{\mathbf{F}}_n^{(s)T} \sum_{j=1}^k {}^{(j)}\Delta \mathbf{d}_n^{(s)}} \leq \varepsilon_c \quad (48)$$

be approximated by a known previous value  $\mathbf{K}_{T_{n-1}}^{(s)}$ . An iterative solver of the Conjugate Gradient (CG) algorithm

(Gill and Murray, 1974) explained in following subsection is employed for defining an increment of traction forces  ${}^{(k)}\Delta \boldsymbol{\lambda}_n$ . After  ${}^{(k)}\Delta \boldsymbol{\lambda}_n$  has been obtained, the increment of displacement vector for each subdomain at current iteration  ${}^{(k)}\Delta \mathbf{d}_n^{(s)}$  can be evaluated by the following equation.

$$\Delta \mathbf{d}_n^{(s)} = \sum_{k=1}^{nk} {}^{(k)}\Delta \mathbf{d}_n^{(s)} \quad (49)$$

The iterative process in the Newton-Raphson algorithm is terminated when the following convergence criterion in each subdomain is satisfied.

Where  $\varepsilon_c$  is a convergence criterion constant. If  $nk$  is a number of converged Newton-Raphson iteration, the increment of displacement vector for current time

$${}_n\Delta \sigma_{ij} = {}_nD_{ijkl}^{\text{ep}} {}_n\Delta \varepsilon_{kl} + {}_n\alpha^2 \delta_{ij} {}_nK_{gf} \delta_{kl} {}_n\Delta \varepsilon_{kl} + {}_n\alpha \delta_{ij} {}_n\Delta \zeta \quad (50)$$

step is defined as follow,

$${}_n\Delta p = {}_nK_{gf} ({}_n\alpha {}_n\Delta \varepsilon_{ii} + {}_n\Delta \zeta) \quad (51)$$

$${}_n\Delta \sigma'_{ij} = {}_n\Delta \sigma_{ij} - \delta_{ij} {}_n\alpha {}_n\Delta p \quad (52)$$

Finally, the dynamic response for each subdomain at current time step can be computed using

$${}_n\alpha = 1 - \frac{\delta_{ij} {}_nD_{ijkl}^{\text{ep}} \delta_{kl}}{3K_g} \quad \text{Equation 37, 38 and 39 for displacement, velocity and acceleration, respectively.} \quad (53)$$

$$\frac{1}{{}_nK_{gf}} = \frac{f}{K_f} + \frac{{}_n\alpha - f}{{}_nK_g} \quad (54)$$

For a given strain increment of the soil grains  ${}_n\Delta \varepsilon_{ii}$  and the volume change of the pore water  ${}_n\Delta \zeta$ , which are evaluated based on the **Equation 8 and 9**, respectively, the increment of the stresses state can be calculated according to the **Equation 4 and 5**. In numerical analysis, the evaluation of these stresses is conducted

as in **Equation 50 to 54**. In these equations a notation  $\Delta$  for denoting the increment is introduced instead of superposed dot used in **Equation 4 and 5**.

$${}_n D_{ijkl}^{ep} = \frac{1}{2} \left( {}_{n-1} D_{ijkl}^{ep} + {}_n D_{ijkl}^{ep} \right) \quad (55)$$

$${}_n \sigma'_{ij} = {}_{n-1} \sigma'_{ij} + {}_n \Delta \sigma'_{ij} \quad (56)$$

The stresses state increment at current are depending on the unknown quantity of the elastoplastic stress-strain relation for soil grains  ${}_n D_{ijkl}^{ep}$ . In order to find the answer of the stresses state increment **equations 50 to 52**, a local iteration has been introduced. For this purpose, the elastoplastic stress-strain relation for the soil grains  ${}_n D_{ijkl}^{ep}$  is approximated as follows (Tanjung, 2002).

A  ${}_{n-1} D_{ijkl}^{ep}$  is the elastoplastic stress-strain relation for the soil grains defined based on the effective stress  ${}_{n-1} \sigma'_{ij}$  on the previous time step (n-1). Superscript  $p$  in the second term of **Equation 55** denotes it elastoplastic stress-strain relation evaluated according to a predicted effective stress below.

The predicted effective stress increment  ${}_n \Delta \sigma'_{ij}$  is approximated based on the value of the effective stress increment on the previous local iteration. Both of elastoplastic stress-strain relations for the soil grains written in the right hand side of **Equation 55** are evaluated using the simplified bounding surface plasticity model previously mentioned. The local iteration process is terminated when a ratio of the norm of the residual of the predicted effective stress increment  ${}_n \Delta \sigma'_{ij}$  to the effective stress increment  ${}_n \Delta \sigma'_{ij}$  goes vanishing.

### 4.3 Parallel computation algorithm

Most of the computations for the Parallel FEM formulations developed in the previous subsection can be performed in the subdomain basis. These computations are identical with the works in the conventional finite

element analysis, such as forming and assembling mass matrix  $\mathbf{M}^{(s)}$ , damping matrix  $\mathbf{D}^{(s)}$ , tangential stiffness matrix  $\mathbf{K}_{Tu}^{(s)}$ , the dynamic loading vector  $\mathbf{F}^{(s)}$  and evaluating **equations 50 to 62**. When these works are assigned to the individual processors in the multi-processors machine, each processor will performs an identical works as above. The different work in each processor is only related to the properties of the considered subdomain. Each processor will analyze a subdomain according to it own input data. When the computations have terminated, each processor will also record it own results. Since each processor conducts the identical works, we only need to develop a single algorithm flow as the usual finite element analysis. The algorithm may be executed in the assigned processor. A special treatment is only required to manage the algorithm for per-

forming the components of matrix-matrix  $\left[ \mathbf{B}^{(s)} \hat{\mathbf{K}}_n^{(s)-1} \mathbf{B}^{(s)T} \right]$

forming the computations with respect to the different input data – related to the different properties of the subdomains – on the different processors. When the processors need to exchange the

since the components of matrix  $\mathbf{B}^{(s)} \hat{\mathbf{K}}_n^{(s)-1} \mathbf{B}^{(s)T}$  are

data with other processors, the processors can explicitly send and/or receive to and/or from other processors. This type of programming model is known as Single Program Multiple Data (SPMD) programming model (Kawamura and Tanjung, 2000, Tanjung, 2002, Baker and Smith,

and the initial residual vector  $\mathbf{r}_0 = \mathbf{B}^{(s)} \hat{\mathbf{K}}_n^{(s)-1} \mathbf{B}^{(s)T} \Delta \mathbf{R}_n^{(s)}$

1987, Kumar, 1994). Another different point to a sequential programming model is only related to the interprocessor communication. This interprocessor communication only takes

$$\begin{aligned} \mathbf{z}_{i-1} &= \bar{\mathbf{A}}_I^{-1} \mathbf{r}_{i-1} \\ \zeta_i &= \mathbf{z}_{i-1}^T \mathbf{r}_{i-1} / \mathbf{z}_{i-2}^T \mathbf{r}_{i-2} \quad (\zeta_1 = 0) \\ \mathbf{s}_i &= \mathbf{z}_{i-1} + \zeta_i \mathbf{s}_{i-1} \quad (\mathbf{s}_0 = \mathbf{z}_0) \\ \mathbf{v}_i &= \mathbf{z}_{i-1}^T \mathbf{s}_{i-1} / \mathbf{s}_i^T \left( \mathbf{B}^{(s)} \hat{\mathbf{K}}_n^{(s)-1} \mathbf{B}^{(s)T} \right) \mathbf{s}_i \quad (57) \\ \lambda_i &= \lambda_{i-1} + \mathbf{v}_i \mathbf{s}_i \\ \mathbf{r}_i &= \mathbf{r}_{i-1} - \mathbf{v}_i \left( \mathbf{B}^{(s)} \hat{\mathbf{K}}_n^{(s)-1} \mathbf{B}^{(s)T} \right) \mathbf{s}_i \end{aligned}$$

place when solves an interface problem written in **Equation 48**.  
As

A term of  $\bar{\mathbf{A}}_I^{-1}$  in the CG algorithms (57) above is mentioned above, the traction vector  ${}^{(k)}\mathbf{Dl}_n$  is defined by converting the boundary value problem into an interface problem as written in **Equation 42**. It problem is



$$\bar{\mathbf{A}}_I^{-1} = \sum_{s=1}^{Ns} \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{K}}_{bb}^{(s)} \end{bmatrix} \quad \begin{array}{l} \text{identical to a minimiz-} \\ \text{ing problem respect} \\ \text{to continuity condition} \end{array} \quad (58)$$

given in Equation 43. In this study, an iterative solver of the Conjugate Gradient (CG) algorithm was used to solve its interface problem. One of the features of the CG method is

in this study – are maintained without modification during iteration. This feature is very convenient to solve the minimizing problem in the parallel machine constructed in the several independent individual processors. Specifically, the algorithm of CG method goes as follow. A superscript  $k$  traction vector  $^{(k)}\Delta\lambda_n$  is temporary omitted for simplification. The initial value of the tractor vector  $\lambda_0$  is given as a null vector;  $\lambda_0 = 0$

The following calculations are iterated until convergence criterion is confirmed (Kawamura and Tanjung, 2000, Tanjung, 2002).

known as a preconditioner. The preconditioner is used to enhance the convergence rate. The form of the preconditioner is written in Equation 58 following the form proposed by Farhat (1991).

The solution of the interface problem requires inter-processor communication such that the data can be exchanged from one processor to others. In fact this exchanged data significantly affects the parallel computing performance, since the communication will delay the calculation works during analysis. There are several patterns of the interprocessor communication commonly used in the parallel algorithms. A simple way is to collect all subdo-

product  $\mathbf{B}^{(s)} \hat{\mathbf{K}}_{T_n}^{(s)-1} \mathbf{B}^{(s)T} \mathbf{s}_i$  in the formulation of

$$\mathbf{v}_i = \mathbf{z}_{i-1}^T \mathbf{r}_{i-1} / \mathbf{s}_i^T \mathbf{B}^{(s)} \hat{\mathbf{K}}_{T_n}^{(s)-1} \mathbf{z}^{(s)T} \mathbf{s}_i \quad \text{in Equation 63}$$

main's contribution in Equation 42 onto another processor, say vector  $\mathbf{y}_k$ , which is calculated as. solves using  $\mathbf{s}_i = \mathbf{B}^{(s)} \hat{\mathbf{K}}_{T_n}^{(s)-1} \mathbf{B}^{(s)T} \mathbf{s}_i$  CG algorithm (58). When the CG has converged, the solution is broadcasted to all proc-

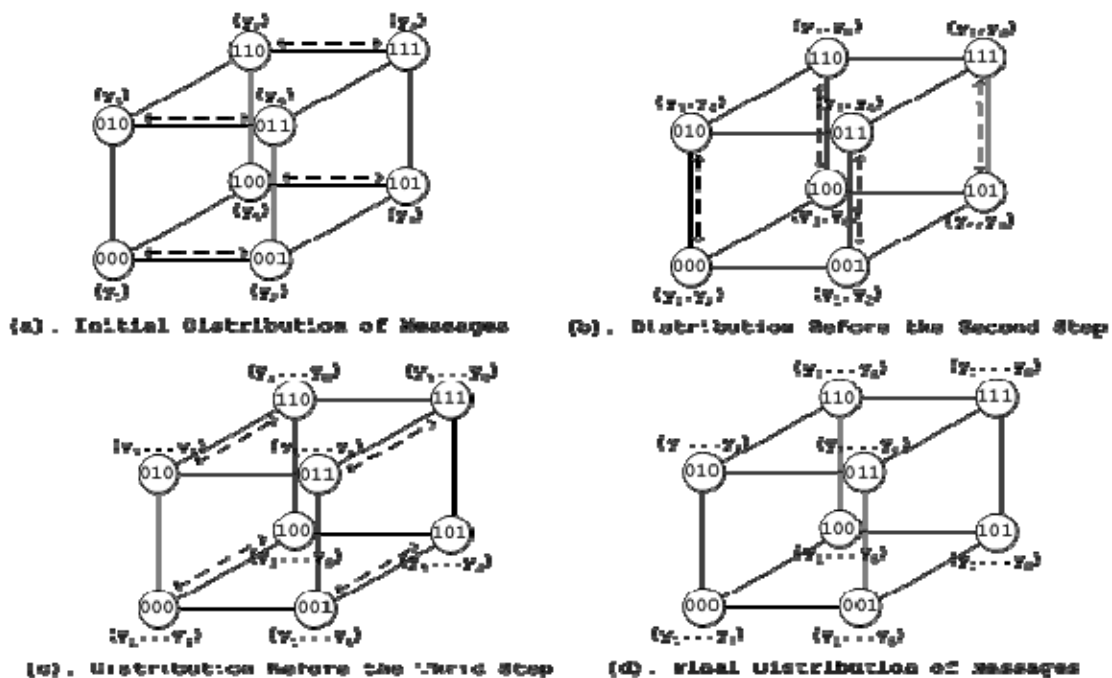


Figure 5. Interprocessor communications on 3D-hypercube

(Source: Tanjung, 2002)

essors. By this way, all processors for analyzing the subdomains of the FEM analysis are only communicating with the processors used to solve the interface problem. Unfortunately, this simple way requires transferring large number of data. This type of interprocessor communication will also reduce the rank of the parallelism since all processors for analyzing the subdomains will idle during solving the interface problem. To improve the simple interprocessor communication above, an interprocessor communication based on the hypercube networking was constructed in this study (Kawamura and Tanjung, 2000, Tanjung, 2002, Kawamura and Tanjung, 2001).

To make clear an understanding of current the interprocessor communication pattern, consider a three-dimensional hypercube scheme shown in **Figure 5** below for instance.

For the first step, maps the subdomains onto the hypercube networking such that each processor is assigned into different active processor. Then, the interprocessor communication is carried out as graphically shown in the **Figure 5**. For example, assume that each vector  $y_k$  in parentheses in the figure denotes the messages, which is required to exchange from one processor to the others. The interprocessor communication starts from the lowest dimension of the hypercube and then proceeds along successively higher dimensions. The processors communicate in pairs with least significant bit of their binary representations. At the termination of communication procedures, each processor will hold the same overall summation result of the vector. Noting that, in the hypercube pattern each processor is only communicate with the neighbours. The neighbours are mapped to hypercube processors whose processor labels differ in exactly one bit position.

Although the CG algorithm requires the interprocessor communication, the algorithm is still amenable to work in the parallel manner. Consider the matrix-vector

as an example. Each processor evaluates the contribu-

Then exchange the values of the vector  $y_k$  follows the proposed interprocessor communication pattern mentioned in the foregoing paragraph to obtain the global values containing the contribution from all subdomains. This example clearly shows the attractive CG

algorithm using the communication pattern mentioned above to solve the interface problem (42).

## 5. Conclusions

Comparing the proposed interprocessor communication pattern based on the hypercube with the simple communication ones, the advantages of the proposed pattern are clearly shown. Each processor is only needed to exchange data in the vector type instead of the matrix/vector in the simple communication pattern. The proposed communication pattern also does not require an additional processor to perform stand alone the CG algorithm. Finally, all of active processors almost never idle since the computation still is conducted in parallel manner even for conducting the CG algorithms.

## References

- Baker, L., Smith, B.J., 1987, *Parallel Programming*, New York: McGraw-Hill.
- Crouch, R.S. and Wolf, J.P., 1994, Unified 3D Critical State Bounding Surface Plasticity Model for Soils incorporating Continues Plasticity Loading Under Cyclic Path, Constitutive Relations, *Int. Journal. Num. Anal. Met., Geomech.* Vol. 18, pp, 735-758.
- Dafalias, Y.F., Bounding Surface Plasticity I, 1986, Mathematical Foundation and Hypoplasticity, *Journal of Eng. Mech. Geomech*, Vol. 112, No. 12, pp, 966-987.
- Farhat, C. and Roux, F.X., 1991, A Method of Finite Element Tearing and Interconnecting and Its Parallel Solution Algorithm, *Int. Journal for Num. Methods Engg.*, Vol. 32, pp, 1205-1227.
- Ghaboussi, A.M. et al., 1973, Seismic Analysis of Earth Dam Reservoir System, *J. Soil Mech. And Found. Div.*, ASCE., Vol. 99, No. SM10, pp. 849-862.
- Gill, P.E. and Murray, W., 1974, *Numerical Methods for Constrained Optimization*, New York: Academic Press.
- Hughes, T.J.R., 1996, *The Finite Element Method Linear Static and Dynamic Finite Element Analysis*, New Jersey: Prentice Hall.
- Kawamura, M. and Tanjung, J., 2000, Parallel FEM Analysis for Dynamic Responses of Saturated Soil Layer, *Journal of Structural Engineering*. JSCE, Vol. 46A, pp. 703-712.