The Decomposed Element-Free Galerkin Method Compared with the Finite Difference Method for Elastic Wave Propagation

Journal:	*Geophysics*
Manuscript ID:	GEO-2007-0178.R2
Manuscript Type:	Geophysical Software and Algorithms
Date Submitted by the Author:	n/a
Complete List of Authors:	Katou, Masafumi; Japan Petroleum Exploration, Exploration Division Matsuoka, Toshifumi; Kyoto Univ., Dept. of Civil and Earth Resource Eng. Mikada, Hitoshi; Kyoto Univ., Dept. of Civil and Earth Resource Eng. Sanada, Yoshinori; Japan Agency for Marine-Earth Science and Technology Ashida, Yuzuru; Environment, Energy, Forestry, and Agriculture Network
Keywords:	2D, wave propagation
Area of Expertise:	Geophysical Software and Algorithms, Seismic Modeling and Wave Propagation

The Decomposed Element-Free Galerkin Method Compared with the Finite Difference Method for Elastic Wave Propagation

Authors: Masafumi Katou[1,2], Toshifumi Matsuoka[1], Hitoshi Mikada[1], Yoshiori Sanada[3], Yuzuru Ashida[4]

1) Dept. of Civil and Earth Resources Eng., Kyoto University, Kyotodaigaku-Katsura, Nishikyo-ku, Kyoto 615-8540, Japan

2) Exploration Division, Japan Petroleum Exploration, Sapia Tower, 1-7-12 Marunouchi, Chiyoda-ku, Tokyo 100-0005, Japan

3) Center of Deep Earth Exploration, Japan Agency for Marine-Earth Science and Technology, 3173-25 Showa-cho, Kanazawa-ku, Yokohama, Kanagawa 236-0001, Japan

4) Environment, Energy, Forestry, and Agriculture Network, 24 Yabusita-cho, Matsubara-dori Shin-machi Nishi-hairu, Shimogyo-ku, Kyoto 600-8448, Japan

Corresponding author: Masafumi Katou

Tel: +81-3-6268-7130

Fax: +81-3-6268-7303

E-mail: masafumi-katou@homail.co.jp

Running title: Decomposed Element-Free Galerkin Method

Abstract

We propose the decomposed element-free Galerkin method (DEFGM) as a modified scheme to resolve shortcomings of memory usage in element-free Galerkin methods (EFGM). The DEFGM decomposes the stiffness matrix in EFGMs into individual schemes and adapts an explicit time-update scheme. In other words, the DEFGM solves elastic wave equation problems by alternately updating the stress-strain relations and the equations of motion as in the staggered-grid finite-difference method (FDM). The DEFGM requires at most twice the memory space, a size comparable to that used by the FDM. In addition, the DEFGM can adopt perfectly matched layer (PML) absorbing boundary conditions as in the case of the FDM. We therefore can make a fair comparison between the DEFGM and the FDM. To confirm that the DEFGM performs as well as the FDM, we compared a two-dimensional DEFGM under PML boundary conditions with an FDM with fourth-order spatial accuracy (FDM4). We compared the DEFGM and FDM4 by using an exact analytical solution of PS reflection waves. The results from the DEFGM were as accurate as those obtained by FDM4. We conducted another comparison by using Lamb's problem under the condition of 8 nodal spaces for the shortest S-wavelength. Remarkably, the DEFGM provided an accurate Rayleigh waveform over a distance of at least 50 wavelengths compared with 5 wavelengths for

FDM4. In this Rayleigh-wave case, the DEFGM with 1-m grid spacing was more accurate than FDM4 with 0.5-m grid spacing. In this comparison, the CPU time used by the DEFGM was less than that used by FDM4. Our results demonstrate that the DEFGM could be a suitable method for numerical simulations of elastic wavefields, especially in cases where a free surface is considered.

1. Introduction

Though many numerical methods have been investigated for solving the elastic wave equation, the finite difference method (FDM) using the staggered-grid scheme (e.g., Virieux, 1986; Graves, 1996) is the most popular because of its simple coding and reasonable accuracy. On the other hand, investigation from various angles of the finite element method (FEM) has been increasing. For example, Komatitsch and Tromp (1999) concluded that the spectral element method (SEM) based on the FEM provides more accurate solutions than the FDM, since the SEM adopts a higher-order polynomial interpolation. With this higher-order polynomial interpolation, Käser and Dumbser (2006) and Dumbser and Käser (2006) demonstrated that the arbitrary high-order derivatives discontinuous Galerkin method (ADER-DG) could handle complex structure problems by employing triangular or tetrahedral meshes. Min et al. (2003) showed that the numerical accuracy of the FEM could be improved by a weighted averaging method over neighboring finite elements.

Belytschko et al. (1994) proposed the element-free Galerkin method (EFGM), which is an FEM with moving least squares (MLS) interpolants. Belytschko et al. (1994) simulated the deformation of fracture phenomena of elastic bodies by solving static equilibria using the EFGM. Lu et al. (1995) advanced the EFGM to fracture dynamics by solving equations of motion. Recently, Jia and Hu (2006) used the EFGM to simulate the propagation of elastic waves. As shown by these examples of fracture mechanics, there is much about mesh-free methods (Liu, 2003) to be investigated in more detail for further use. Therefore, FEM-based methodologies need to be reevaluated for future application to elastic wave propagation problems.

The EFGM performs with high accuracy even using a low-order (second-order at most) polynomial interpolation base function when static or fracture problems are solved (Belytschko et al., 1994). Although such high performance is expected in the case of wave propagation problems, it is difficult to apply the EFGM to large dynamic problems since it uses a stiffness matrix. While these earlier studies adopted a stiffness matrix formula, we need to handle this large matrix in a numerical scheme. In fact, the computations in Jia and Hu (2006) handled at most 41 × 41 nodal points. Therefore, the computational model is applicable only to small models because of memory restrictions.

Many authors have tried to avoid the utilization of the stiffness matrix in the standard FEM case (Koketsu et al., 2004; Ma et al., 2004; Ichimura et al, 2007) in which a second-order system of wave equations is used. Since perfectly matched layer (PML) boundary conditions for a second-order system are far more complicated than for a first-order system (Komatitsch and Tromp, 2003), we tried a first-order velocity-stress formulation of the elastic wave equation (e.g., Collino and Tsogka, 2001) in this study as in the case for the staggered-grid FDM for simplified PML implementation. We used the EFGM with third-order spatial accuracy for enhanced accuracy as compared to the FDM with fourth-order spatial accuracy (FDM4).

Applying this set of ideas to the EFGM, we call this new methodology the decomposed element-free Galerkin method (DEFGM). This methodology could reduce memory usage in the EFGM and allow a fair comparison between DEFGM and FDM4 in terms of memory usage. In this paper, we first introduce DEFGM methodology without using a large stiffness matrix and show how PML boundary conditions are adopted in the DEFGM scheme. We next discuss the CPU time requirements of this methodology. Finally, we examine the results of solutions for PS reflection waves and Lamb's problem by using the DEFGM and FDM4. Remarkably, the DEFGM provides accurate Rayleigh waveforms for a distance of at least 50 wavelengths while FDM4 is able to do the same for only 5 wavelengths. We also found that the DEFGM with 1-m nodal spacing is more accurate than FDM4 with 0.5-m grid spacing.

2. Shape function and time update schemes for stress-strain relations

The original computational procedure of the EFGM was introduced by Belytschko et al. (1994), who used a stiffness matrix formula. In this paper, we avoid the stiffness matrix formulation and propose a new numerical scheme without a large stiffness matrix. This technique for decomposing the stiffness matrix into individual schemes makes it possible to handle as large a number of grids as in the FDM.

In this method, a coupled first-order velocity-stress formulation of the elastic wave equation is solved. The DEFGM therefore solves elastic wave propagation problems by alternatively updating stress-strain relations and equations of motion. In this section, a shape function that interpolates particle velocity by the EFGM, the stress-strain relation, and the time update scheme are presented.

Interpolating the shape function by the moving least squares method

The velocity vector and the stress tensor are arranged in a rectangular element as in Figure 1. (x_0 , z_0) is the central position of the element, and Δx and Δz are the nodal spacings in the x- and z-directions, respectively. There are 3 × 3 Gauss-Legendre (GL) integral points (i = i, ii, …, ix) shown by filled squares. The nodes (j = I, II, …, IX) are illustrated by open circles. When particle velocity vectors are given at these nine nodes, the stress tensor can be evaluated at the nine GL integral points by multiplying a coefficient matrix by the velocity vectors. This coefficient matrix is determined by the formation of nine GL points and nine nodal points, and is obtained as follows.

First, we propose the following base vector for the shape function:

$$\mathbf{P}^{\mathrm{T}} = [1, x, z, xz, x^2, z^2, x^2z^2]. \qquad (2.1)$$

Next, we adopt the following weight function:

$$w_i(r_{ij}) = \begin{cases} n\dfrac{r_{ij}}{\mathrm{R}_i}\left(1-\dfrac{r_{ij}}{\mathrm{R}_i}\right)^{n-1} + \left(1-\dfrac{r_{ij}}{\mathrm{R}_i}\right)^{n} & 0 \le r_{ij} \le \mathrm{R}_i \\ 0 & r_{ij} > \mathrm{R}_i \end{cases}, \quad (2.2)$$

where r_{ij} is the distance between each pair of GL points and nodal points, and $r_{ij} = \sqrt{\left(x_j - x_i\right)^2 + \left(z_j - z_i\right)^2}$. R_i is the affection radius for each GL point, and n is an arbitrary natural number. This function is useful for arranging the inflection points in a simple way by selecting the arbitrary number n. Figure 2 shows this weight function (2.2) with n = 6. Equation (2.2) with n = 4 was a popular weight function among earlier works (e.g. Beissel and Belytschko, 1996; Liu, 2003; Brighenti, 2005). Although the choice of base vector and weight function controls numerical accuracy in the EFGM, we choose them because they are simple to introduce and provide sufficient accuracy (see section 7).

When a GL point i in an element is located at a point (x_i, z_i), the coefficients of the interpolated particle velocity,

$$v_i = a_1 + a_2 x_i + a_3 z_i + a_4 x_i z_i + a_5 x_i^2 + a_6 z_i^2 + a_7 x_i^2 z_i^2 , \quad (2.3)$$

satisfy the following equation:

$$W_i B A = W_i V , \qquad (2.4)$$

where

$$\mathrm{diag}\left[W_i\right] = \left[w_i\left(r_{i\mathrm{I}}\right) \quad w_i\left(r_{i\mathrm{II}}\right) \quad \cdots \quad w_i\left(r_{i\mathrm{IX}}\right) \right]^{\mathrm{T}} , \qquad (2.5)$$

$$B = \begin{bmatrix} 1 & x_0-\Delta x & z_0-\Delta z & (x_0-\Delta x)(z_0-\Delta z) & (x_0-\Delta x)^2 & (z_0-\Delta z)^2 & (x_0-\Delta x)^2(z_0-\Delta z)^2 \\ 1 & x_0 & z_0-\Delta z & x_0(z_0-\Delta z) & x_0^2 & (z_0-\Delta z)^2 & x_0^2(z_0-\Delta z)^2 \\ 1 & x_0-\Delta x & z_0 & (x_0-\Delta x)z_0 & (x_0-\Delta x)^2 & z_0^2 & (x_0-\Delta x)^2 z_0^2 \\ 1 & x_0 & z_0 & x_0 z_0 & x_0^2 & z_0^2 & x_0^2 z_0^2 \\ 1 & x_0+\Delta x & z_0-\Delta z & (x_0+\Delta x)(z_0-\Delta z) & (x_0+\Delta x)^2 & (z_0-\Delta z)^2 & (x_0+\Delta x)^2(z_0-\Delta z)^2 \\ 1 & x_0-\Delta x & z_0+\Delta z & (x_0-\Delta x)(z_0+\Delta z) & (x_0-\Delta x)^2 & (z_0+\Delta z)^2 & (x_0-\Delta x)^2(z_0+\Delta z)^2 \\ 1 & x_0+\Delta x & z_0 & (x_0+\Delta x)z_0 & (x_0+\Delta x)^2 & z_0^2 & (x_0+\Delta x)^2 z_0^2 \\ 1 & x_0 & z_0+\Delta z & x_0(z_0+\Delta z) & x_0^2 & (z_0+\Delta z)^2 & x_0^2(z_0+\Delta z)^2 \\ 1 & x_0+\Delta x & z_0+\Delta z & (x_0+\Delta x)(z_0+\Delta z) & (x_0+\Delta x)^2 & (z_0+\Delta z)^2 & (x_0+\Delta x)^2(z_0+\Delta z)^2 \end{bmatrix} ,$$

(2.6)

$$A = \left[a_1 \quad a_2 \quad \cdots \quad a_7 \right]^{\mathrm{T}} , \quad (2.7)$$

$$V=\begin{bmatrix} v(\mathrm{I}) \\ v(\mathrm{II}) \\ v(\mathrm{III}) \\ v(\mathrm{IV}) \\ v(\mathrm{V}) \\ v(\mathrm{VI}) \\ v(\mathrm{VII}) \\ v(\mathrm{VIII}) \\ v(\mathrm{IX}) \end{bmatrix}=\begin{bmatrix} v(x_0-\Delta x, z_0-\Delta z) \\ v(x_0, z_0-\Delta z) \\ v(x_0-\Delta x, z_0) \\ v(x_0, z_0) \\ v(x_0+\Delta x, z_0-\Delta z) \\ v(x_0-\Delta x, z_0+\Delta z) \\ v(x_0+\Delta x, z_0) \\ v(x_0, z_0+\Delta z) \\ v(x_0+\Delta x, z_0+\Delta z) \end{bmatrix}, \qquad (2.8)$$

where T denotes the matrix transpose. Solving equation (2.4) by using the moving least squares (MLS) method gives

$$A=\left[B^{\mathrm{T}}W_iB\right]^{-1}B^{\mathrm{T}}W_iV\,. \qquad (2.9)$$

Thus equation (2.3) becomes

$$v_i=P_iV\,, \qquad (2.10)$$

where

$$P_i=\begin{bmatrix}1 & x_i & z_i & x_iz_i & x_i^2 & z_i^2 & x_i^2z_i^2\end{bmatrix}\left[B^{\mathrm{T}}W_iB\right]^{-1}B^{\mathrm{T}}W_i\,. \qquad (2.11)$$

All P_i vectors for every GL point form the following coefficient matrix:

$$\Phi=\begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_9 \end{bmatrix}=\begin{bmatrix} & \vdots & & \\ \cdots & \phi_{ij} & \cdots & \cdots \\ & \vdots & & \\ & \vdots & & \end{bmatrix}. \qquad (2.12)$$

This ϕ_{ij} is known as a shape function in the EFGM.

Partial derivatives of the shape function

The partial derivatives of the particle velocity v_i at each GL point can be calculated using ϕ_{ij} as follows:

$$\frac{\partial v_i}{\partial x} = \frac{\partial P_i}{\partial x} V \,, \quad (2.13)$$

$$\frac{\partial v_i}{\partial z} = \frac{\partial P_i}{\partial z} V \,, \quad (2.14)$$

where

$$\begin{aligned}\frac{\partial P_i}{\partial x} = & \begin{bmatrix} 0 & 1 & 0 & z_i & 2x_i & 0 & 2x_i z_i^2 \end{bmatrix} \left[B^{\mathrm{T}} W_i B \right]^{-1} B^{\mathrm{T}} W_i \\ & + \begin{bmatrix} 1 & x_i & z_i & x_i z_i & x_i^2 & z_i^2 & x_i^2 z_i^2 \end{bmatrix} \frac{\partial \left[B^{\mathrm{T}} W_i B \right]^{-1}}{\partial x} B^{\mathrm{T}} W_i \,, \qquad (2.15) \\ & + \begin{bmatrix} 1 & x_i & z_i & x_i z_i & x_i^2 & z_i^2 & x_i^2 z_i^2 \end{bmatrix} \left[B^{\mathrm{T}} W_i B \right]^{-1} B^{\mathrm{T}} \frac{\partial W_i}{\partial x} \end{aligned}$$

$$\begin{aligned}\frac{\partial P_i}{\partial z} = & \begin{bmatrix} 0 & 0 & 1 & x_i & 0 & 2z_i & 2x_i^2 z_i \end{bmatrix} \left[B^{\mathrm{T}} W_i B \right]^{-1} B^{\mathrm{T}} W_i \\ & + \begin{bmatrix} 1 & x_i & z_i & x_i z_i & x_i^2 & z_i^2 & x_i^2 z_i^2 \end{bmatrix} \frac{\partial \left[B^{\mathrm{T}} W_i B \right]^{-1}}{\partial z} B^{\mathrm{T}} W_i \,, \qquad (2.16) \\ & + \begin{bmatrix} 1 & x_i & z_i & x_i z_i & x_i^2 & z_i^2 & x_i^2 z_i^2 \end{bmatrix} \left[B^{\mathrm{T}} W_i B \right]^{-1} B^{\mathrm{T}} \frac{\partial W_i}{\partial z} \end{aligned}$$

$$\frac{\partial \left[B^{\mathrm{T}} W_i B \right]^{-1}}{\partial k} = -\left[B^{\mathrm{T}} W_i B \right]^{-1} \left[B^{\mathrm{T}} \frac{\partial W_i}{\partial k} B \right] \left[B^{\mathrm{T}} W_i B \right]^{-1} \quad (k = x, z). \quad (2.17)$$

Dynamic problems, such as wave propagation phenomena, generally assume that the displacements caused by elastic waves are negligible in infinitesimal displacement theory, therefore $\partial P_i / \partial x$ and $\partial P_i / \partial z$ can be considered as constant throughout the simulation and need only be computed once after the geometrical parameters Δx and Δz

are given.

Parameters for the shape function

For this paper, all simulations adopted the following parameters: the elemental volume was assumed to be 2Δx × 2Δz, the nodal spacing Δx and Δz were the same (Δx = Δz), and the affection radius R_i for each GL point was set to be

$$\mathrm{R}_i = \begin{cases} 0.8\times 2\Delta x \ \ (\text{for } i = \text{iv}) \\ 1.1\times 2\Delta x \ \ (\text{for } i = \text{ii, iii, vii, viii}) \\ 1.3\times 2\Delta x \ \ (\text{for } i = \text{i, v, vi, ix}) \end{cases}. \qquad (2.18)$$

These values were set to a distance that is a little longer than the distance between the farthest pair of nodal points and GL points. From computational trials, we found that the set of values in equation (2.18) performs better than

$$\mathrm{R}_i = \begin{cases} 0.7\times 2\Delta x \ \ (\text{for } i = \text{iv}) \\ 1.0\times 2\Delta x \ \ (\text{for } i = \text{ii, iii, vii, viii}) \\ 1.2\times 2\Delta x \ \ (\text{for } i = \text{i, v, vi, ix}) \end{cases} \text{ or } \mathrm{R}_i = \begin{cases} 0.9\times 2\Delta x \ \ (\text{for } i = \text{iv}) \\ 1.2\times 2\Delta x \ \ (\text{for } i = \text{ii, iii, vii, viii}) \\ 1.4\times 2\Delta x \ \ (\text{for } i = \text{i, v, vi, ix}) \end{cases}.$$

The value of n in the weight function, equation (2.2), is set to n = 5. In order to stably compute ϕ_{ij}, we use x_0 = −Δx and z_0 = −Δz. We chose n = 6 as the best value after performing computational trials for n = 3, 4, $\cdots$, 7. There is definitely a possibility that the accuracy could be increased by modifying equations (2.1), (2.2), R_i, or n. The results from some other choices are shown in section 7.

Time update schemes for the stress–strain relations

As is well known, the velocity-stress formulation of the elastic wave equation comprises two sets of equations: stress-strain relations and the equations of motion. The stress-strain relations are given as follows:

$$\frac{\partial \sigma_{xx}}{\partial t} = (\lambda + 2\mu)\frac{\partial v_x}{\partial x} + \lambda \frac{\partial v_z}{\partial z}, \quad (2.19a)$$

$$\frac{\partial \sigma_{zz}}{\partial t} = \lambda \frac{\partial v_x}{\partial x} + (\lambda + 2\mu)\frac{\partial v_z}{\partial z}, \quad (2.19b)$$

$$\frac{\partial \sigma_{xz}}{\partial t} = \mu \left(\frac{\partial v_x}{\partial z} + \frac{\partial v_z}{\partial x} \right), \quad (2.19c)$$

where λ and μ are Lame's moduli; σ_{xx}, σ_{zz}, and σ_{xz} are the stresses; and v_x and v_z are the particle velocities. On a GL point i at a specific time t, by employing explicit discretization of second-order accuracy in time and interpolation of particle velocity, equations (2.19a-c) become

$$\frac{\sigma_{xx}^{t+\Delta t/2}(x_i, z_i) - \sigma_{xx}^{t-\Delta t/2}(x_i, z_i)}{\Delta t} = (\lambda + 2\mu)\frac{\partial P_i}{\partial x} V_x^t + \lambda \frac{\partial P_i}{\partial z} V_z^t, \quad (2.20a)$$

$$\frac{\sigma_{zz}^{t+\Delta t/2}(x_i, z_i) - \sigma_{zz}^{t-\Delta t/2}(x_i, z_i)}{\Delta t} = \lambda \frac{\partial P_i}{\partial x} V_x^t + (\lambda + 2\mu)\frac{\partial P_i}{\partial z} V_z^t, \quad (2.20b)$$

$$\frac{\sigma_{xz}^{t+\Delta t/2}(x_i, z_i) - \sigma_{xz}^{t-\Delta t/2}(x_i, z_i)}{\Delta t} = \mu \left(\frac{\partial P_i}{\partial z} V_x^t + \frac{\partial P_i}{\partial x} V_z^t \right), \quad (2.20c)$$

where Δt is the sampling time length, the superscript is the computing time, and

$$V_k = \begin{bmatrix} v_k(\mathrm{I}) \\ v_k(\mathrm{II}) \\ v_k(\mathrm{III}) \\ v_k(\mathrm{IV}) \\ v_k(\mathrm{V}) \\ v_k(\mathrm{VI}) \\ v_k(\mathrm{VII}) \\ v_k(\mathrm{VIII}) \\ v_k(\mathrm{IX}) \end{bmatrix} = \begin{bmatrix} v_k(x_0-\Delta x, z_0-\Delta z) \\ v_k(x_0, z_0-\Delta z) \\ v_k(x_0-\Delta x, z_0) \\ v_k(x_0, z_0) \\ v_k(x_0+\Delta x, z_0-\Delta z) \\ v_k(x_0-\Delta x, z_0+\Delta z) \\ v_k(x_0+\Delta x, z_0) \\ v_k(x_0, z_0+\Delta z) \\ v_k(x_0+\Delta x, z_0+\Delta z) \end{bmatrix}, \; (k=x,\, z). \qquad (2.21)$$

3. Equations of motion

In this section, the equations of motion are discussed. Figure 3 shows an elastic body consisting of nine elements. Open circles show the nodal points. The equations of motion are as follows:

$$\rho\frac{\partial v_k}{\partial t} = f_k + \frac{\partial \sigma_{kx}}{\partial x} + \frac{\partial \sigma_{kz}}{\partial z}, \qquad (k=x,\, z) \qquad (3.1)$$

where f_k is a component of an external acceleration vector and ρ is the material density. These partial differential equations give the following weighted residual equations:

$$\int_\Omega \varphi\left(\rho\frac{\partial v_k}{\partial t} - f_k - \frac{\partial \sigma_{kx}}{\partial x} - \frac{\partial \sigma_{kz}}{\partial z}\right) d\Omega = 0. \qquad (3.2)$$

In the Galerkin method, the shape function ϕ_{ij} introduced in the previous section is used as the weight function φ. By using integration by parts, equation (3.2) becomes

$$\int_\Omega \varphi\rho\frac{\partial v_k}{\partial t} d\Omega = \int_\Omega \varphi f_k d\Omega + \int_\Gamma \varphi\left(\tau_{kx}n_x + \tau_{kz}n_z\right) d\Gamma - \int_\Omega \left(\frac{\partial \varphi}{\partial x}\sigma_{kx} + \frac{\partial \varphi}{\partial z}\sigma_{kz}\right) d\Omega, \qquad (3.3)$$

where τ_{kx} and τ_{kz} are the components of the external stress tensor on the surface (Γ), and

n_x and n_x denote the components of the normal vector.

Lumped mass matrix

By employing GL integration, the left-hand side of equation (3.3) is discretized in space as

$$\int_{\Omega}\varphi\rho\frac{\partial v_k}{\partial t}d\Omega=\frac{\partial}{\partial t}\sum_{\Omega}\left[\left\{4\Delta x\Delta z\rho\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[P_i^{\mathrm{T}}q_iP_i\right]\right\}V_k\left(t\right)\right], \quad (3.4)$$

where q_i is the weight value for each GL integration point and $4\Delta x\Delta z$ represents the volume of a single element. The treatment of $\sum\limits_{\Omega}$ as a symbolic calculation in the DEFGM is explained by the later equations (3.11) to (3.13).

We introduce the unit mass matrix M as follows:

$$M=4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[P_i^{\mathrm{T}}q_iP_i\right]=\begin{bmatrix} & \vdots & & \\ \cdots & m_{j_1j_2} & \cdots & \cdots \\ & \vdots & & \\ & \vdots & & \end{bmatrix}, (j_1, j_2 = \mathrm{I}, \mathrm{II}, \ldots, \mathrm{IX}). \quad (3.5)$$

To solve the dynamic problem explicitly, we employ the following lumped mass matrix $\bar{M}$:

$$\mathrm{diag}\left[\bar{M}\right]=\left[\sum_{j_2=\mathrm{I}}^{\mathrm{IX}}m_{\mathrm{I}j_2} \quad \sum_{j_2=\mathrm{I}}^{\mathrm{IX}}m_{\mathrm{II}j_2} \quad \cdots \quad \sum_{j_2=\mathrm{I}}^{\mathrm{IX}}m_{\mathrm{IX}j_2}\right]=\left[\bar{m}_{\mathrm{I}} \quad \bar{m}_{\mathrm{II}} \quad \cdots \quad \bar{m}_{\mathrm{IX}}\right]. \quad (3.6)$$

Thus equation (3.4) becomes

$$\int_{\Omega} \varphi \rho \frac{\partial v_k}{\partial t} d\Omega = \frac{\partial}{\partial t} \sum_{\Omega} \left[\bar{M} V_k \left(t \right) \right]. \qquad (3.7)$$

Internal stress

By using the shape function, the third term on the right-hand side of equation (3.3) is discretized into the following:

$$\int_{\Omega} \left(\frac{\partial \varphi}{\partial x} \sigma_{kx} + \frac{\partial \varphi}{\partial z} \sigma_{kz} \right) d\Omega = \sum_{\Omega} \left[4 \Delta x \Delta z \sum_{i=\mathrm{i}}^{\mathrm{ix}} \left[q_i \left\{ \left(\frac{\partial P_i}{\partial x} \right)^{\mathrm{T}} \sigma_{kx} \left(x_i, z_i \right) + \left(\frac{\partial P_i}{\partial z} \right)^{\mathrm{T}} \sigma_{kz} \left(x_i, z_i \right) \right\} \right] \right].$$

$$(3.8)$$

External forces

The first and second terms on the right-hand side of equation (3.3) show the external acceleration and stress. They become equivalent forces on the node (F_k) by the shape function, thus

$$\int_{\Omega} \varphi f_k d\Omega + \int_{\Gamma} \varphi \left(\tau_{kx} n_x + \tau_{kz} n_z \right) d\Gamma = \sum_{\Omega} F_k . \qquad (3.9)$$

Time update schemes for the equations of motion

Using the discretization of equations (3.7) to (3.9), equation (3.3) becomes

$$\frac{\partial}{\partial t}\sum_{\Omega}\left[\bar{M}V_k\left(t\right)\right]-\sum_{\Omega}F_k=-\sum_{\Omega}\left[4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\left(\frac{\partial P_i}{\partial x}\right)^{\mathrm{T}}\sigma_{kx}\left(x_i,z_i\right)+\left(\frac{\partial P_i}{\partial z}\right)^{\mathrm{T}}\sigma_{kz}\left(x_i,z_i\right)\right\}\right]\right].$$

(3.10)

Using explicit discretization of second-order accuracy in time, equation (3.10) becomes

$$\begin{aligned}&\frac{1}{\Delta t}\sum_{\Omega}\left[\bar{M}\left(V_k^{t+\Delta t/2}-V_k^{t-\Delta t/2}\right)\right]\\&=\sum_{\Omega}F_k^t\left(t\right)-\sum_{\Omega}\left[4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\left(\frac{\partial P_i}{\partial x}\right)^{\mathrm{T}}\sigma_{kx}^t\left(x_i,z_i\right)+\left(\frac{\partial P_i}{\partial z}\right)^{\mathrm{T}}\sigma_{kz}^t\left(x_i,z_i\right)\right\}\right]\right].\end{aligned}\qquad(3.11)$$

Based on the elemental arrangement of Figure 3, for example, near Ω_5 the particle velocity is shared by neighboring elements on nodal points as follows:

$$V\left(\mathrm{I},\Omega_5\right)=V\left(\mathrm{V},\Omega_4\right)=V\left(\mathrm{VI},\Omega_2\right)=V\left(\mathrm{IX},\Omega_1\right),$$

$$V\left(\mathrm{II},\Omega_5\right)=V\left(\mathrm{VIII},\Omega_2\right),\ \ V\left(\mathrm{III},\Omega_5\right)=V\left(\mathrm{VII},\Omega_4\right).\qquad(3.12)$$

Thus, the summation $\sum_{\Omega}$ in equation (3.11) is the most complex procedure in DEFGM computation and is expressed by the following four patterns:

$$\left(\bar{m}_{\mathrm{I}}\left(\Omega_5\right)+\bar{m}_{\mathrm{V}}\left(\Omega_4\right)+\bar{m}_{\mathrm{VI}}\left(\Omega_2\right)+\bar{m}_{\mathrm{IX}}\left(\Omega_1\right)\right)\frac{v_k^{t+\Delta t/2}\left(\mathrm{I},\Omega_5\right)-v_k^{t-\Delta t/2}\left(\mathrm{I},\Omega_5\right)}{\Delta t}=F_k^t\left(\mathrm{I},\Omega_5\right)$$

$$-4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\frac{\partial\phi_{i\mathrm{I}}\left(\Omega_5\right)}{\partial x}\sigma_{kx}^t\left(x_i,z_i,\Omega_5\right)+\frac{\partial\phi_{i\mathrm{I}}\left(\Omega_5\right)}{\partial z}\sigma_{kz}^t\left(x_i,z_i,\Omega_5\right)\right\}\right]$$

$$-4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\frac{\partial\phi_{i\mathrm{V}}\left(\Omega_4\right)}{\partial x}\sigma_{kx}^t\left(x_i,z_i,\Omega_4\right)+\frac{\partial\phi_{i\mathrm{V}}\left(\Omega_4\right)}{\partial z}\sigma_{kz}^t\left(x_i,z_i,\Omega_4\right)\right\}\right]$$

$$-4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\frac{\partial\phi_{i\mathrm{VI}}(\Omega_2)}{\partial x}\sigma_{kx}^t(x_i,z_i,\Omega_2)+\frac{\partial\phi_{i\mathrm{VI}}(\Omega_2)}{\partial z}\sigma_{kz}^t(x_i,z_i,\Omega_2)\right\}\right]$$

$$-4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\frac{\partial\phi_{i\mathrm{IX}}(\Omega_1)}{\partial x}\sigma_{kx}^t(x_i,z_i,\Omega_1)+\frac{\partial\phi_{i\mathrm{IX}}(\Omega_1)}{\partial z}\sigma_{kz}^t(x_i,z_i,\Omega_1)\right\}\right], \qquad (3.13a)$$

$$\left(\bar{m}_{\mathrm{II}}(\Omega_5)+\bar{m}_{\mathrm{VIII}}(\Omega_2)\right)\frac{v_k^{t+\Delta t/2}(\mathrm{II},\Omega_5)-v_k^{t-\Delta t/2}(\mathrm{II},\Omega_5)}{\Delta t}=F_k^t(\mathrm{II},\Omega_5)$$

$$-4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\frac{\partial\phi_{i\mathrm{II}}(\Omega_5)}{\partial x}\sigma_{kx}^t(x_i,z_i,\Omega_5)+\frac{\partial\phi_{i\mathrm{II}}(\Omega_5)}{\partial z}\sigma_{kz}^t(x_i,z_i,\Omega_5)\right\}\right]$$

$$-4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\frac{\partial\phi_{i\mathrm{VIII}}(\Omega_2)}{\partial x}\sigma_{kx}^t(x_i,z_i,\Omega_2)+\frac{\partial\phi_{i\mathrm{VIII}}(\Omega_2)}{\partial z}\sigma_{kz}^t(x_i,z_i,\Omega_2)\right\}\right], \qquad (3.13b)$$

$$\left(\bar{m}_{\mathrm{III}}(\Omega_5)+\bar{m}_{\mathrm{VII}}(\Omega_4)\right)\frac{v_k^{t+\Delta t/2}(\mathrm{III},\Omega_5)-v_k^{t-\Delta t/2}(\mathrm{III},\Omega_5)}{\Delta t}=F_k^t(\mathrm{III},\Omega_5)$$

$$-4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\frac{\partial\phi_{i\mathrm{II}}(\Omega_5)}{\partial x}\sigma_{kx}^t(x_i,z_i,\Omega_5)+\frac{\partial\phi_{i\mathrm{II}}(\Omega_5)}{\partial z}\sigma_{kz}^t(x_i,z_i,\Omega_5)\right\}\right]$$

$$-4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\frac{\partial\phi_{i\mathrm{VIII}}(\Omega_4)}{\partial x}\sigma_{kx}^t(x_i,z_i,\Omega_4)+\frac{\partial\phi_{i\mathrm{VIII}}(\Omega_4)}{\partial z}\sigma_{kz}^t(x_i,z_i,\Omega_4)\right\}\right], \qquad (3.13c)$$

$$\bar{m}_{\mathrm{IV}}(\Omega_5)\frac{v_k^{t+\Delta t/2}(\mathrm{IV},\Omega_5)-v_k^{t-\Delta t/2}(\mathrm{IV},\Omega_5)}{\Delta t}=F_k^t(\mathrm{IV},\Omega_5)$$

$$-4\Delta x\Delta z\sum_{i=\mathrm{i}}^{\mathrm{ix}}\left[q_i\left\{\frac{\partial\phi_{i\mathrm{IV}}(\Omega_5)}{\partial x}\sigma_{kx}^t(x_i,z_i,\Omega_5)+\frac{\partial\phi_{i\mathrm{IV}}(\Omega_5)}{\partial z}\sigma_{kz}^t(x_i,z_i,\Omega_5)\right\}\right]. \qquad (3.13d)$$

In summary, the practical computational procedure is as follows: at a specific time step, the stress tensor is updated from the particle velocity vector by using equations (2.20a–c). Then the particle velocity vector is updated from the stress tensor by using equations (3.13a-d). These two alternating update processes are repeated for the required number of time steps. We call this methodology the decomposed element-free Galerkin method (DEFGM). Figure 4 shows the flow of DEFGM computation.

4. Stability conditions

Before applying this proposed scheme to realistic subsurface models, we first investigate its stability conditions.

The image method by Levander (1988) is widely used in the FDM framework for expressing a flat free surface, and at least 8 grid spaces are assured for the shortest S-wavelength (Bohlen and Saenger, 2006). We conducted two tests for this paper. The first was determining the solution for a PS reflected wave, which was conducted under the condition of 4 to 8 nodal spaces for the shortest S-wavelength. The second test was solving a Rayleigh wave, which was conducted under the condition of 8 nodal spaces for the shortest S-wavelength.

When $\Delta x = \Delta z$, the sampling time step Δt should be dominated by

$$\Delta t < \frac{c\Delta x}{\max\{V_p\}}, \qquad (4.9)$$

where c is the Courant number and $\max\{V_p\}$ is the maximum P-wave velocity in the medium. We determined experimentally that DEFGM requires c = 0.80 or less. This is the same value as in Koketsu et al., (2004) and it does not change even if the weight function is changed.

5. Computation memory and time requirements

Finite difference method

We adopted a fourth-order standard staggered-grid scheme (FDM4) from Levander (1988). Although a rotated staggered-grid scheme is better than a standard one for a model comprising a topographic surface (Bohlen and Saenger, 2006), we used a standard one because we considered a flat free surface in our investigation of the basic accuracy of the DEFGM. In FDM4, we chose a flat free surface boundary by the image method (Levander, 1988); Figure 5 shows a schematic of our FDM4 grid arrangement and the strategy for the free surface. In Figure 5, Δx and Δz are the grid spacing for the x- and z-directions, respectively. The FDM4 grid spacing is the same parameter as the DEFGM nodal spacing.

Memory requirements

Table 1 shows the general array sizes for the DEFGM and FDM4. In the DEFGM case, when (2nx+1) × (2nz+1) nodal points are evaluated, an array size of nx × nz is required for λ and μ; nx × nz × 9 for σ_{xx}, σ_{zz}, and σ_{xz}; and (2nx + 1) × (2nz + 1) for v_x, v_z, and $\bar{M}$. In the case where a stiffness matrix is used, an array size of 25 × 2 × (2nx + 1) ×(2nz + 1) × 2 is required (= 25 neighboring nodes × 2 components ×(2nx + 1) ×(2nz + 1) total nodes × 2 components).

In the DEFGM numerical scheme configuration, the number of nodal points used to evaluate the particle velocity is the same as in FDM4. However, the number of grid points used to evaluate the stress tensor becomes 9/4 times greater in comparison with FDM4 since these grid points are used for the GL integration (Figure 1). This means that the DEFGM requires at most twice the memory space of FDM4.

Time requirements

The schemes for applying PML to the DEFGM are shown in Appendix A. We used directional splitting for all calculation space even if there was a non-PML area. Therefore, the number of PML layers was not a function of the CPU time.

FDM4 without PML took only about 26 s on a Xeon 3.0 GHz PC when we employed 1000 time steps and 401 × 401 nodal points; on the other hand, the DEFGM without PML needed 1 min 16 s. Table 2 summarizes the calculation times. The values in square brackets are the ratios of the calculation time with respect to the FDM4 time. The DEFGM required 2.9 times the calculation time of FDM4. Although the calculation time of FDM4 became 5.8 times greater when applying PML, that of the DEFGM became about 15 times greater.

Next, we calculated in the same physical space using a smaller nodal spacing. The model consisted of 2000 time steps and 801 × 801 nodal points. When we did not use PML, the DEFGM (1 min 16 s) was faster than FDM4 (4 min 34 s). When we used PML, the DEFGM (18 min 26 s) was faster than FDM4 (30 min 50 s).

6. PS Reflected wave

In the field of exploration geophysics, the reflected wave contains important information. The upper left of Figure 6 shows the calculation model. The model comprises the interface between two elastic media; the upper layer has a P-wave velocity of Vp = 2000 m/s, S-wave velocity of Vs = 1000 m/s, and material density of ρ

= 1500 kg/m^3 and the lower layer has Vp = 2500 m/s, Vs = 1500 m/s, and ρ = 1900 kg/m^3. A 20-m-thick PML is applied to all four sides of the numerical model. There are 401 × 401 nodal points in the model, and the nodal spacing is $\Delta x = \Delta z = 1$ m, which defines a 400 × 400 m calculation space. The compressional source is located at (100 m, 100 m). The Ricker wavelet (second-order derivative of the Gaussian function) generates various peak frequencies including 50, 66, 80, and 100 Hz and a peak amplitude of 1 Pa/m. For example, in the case of the 50 Hz peak frequency, about 125 Hz becomes the maximum frequency component for this implementation. The minimum wave propagation velocity for this model is 1000 m/s; therefore, 8 m is the shortest wavelength and 8 nodal spaces are assured for the wave. The upper right, lower left, and lower right snapshots in Figure 6 show the z-component of the particle velocity at 0.1, 0.14, and 0.18 s, respectively. The PML works effectively.

We obtained an exact waveform from EXE2DELEL from the Spice homepage (http://www.spice-rtn.org). Figure 7 shows a comparison of the waveforms calculated by the DEFGM and FDM4. The analytical solution (thick black line) is plotted against the numerical one (thin gray line) obtained by the DEFGM and FDM4. You can see that both the DEFGM and FDM4 provide good resolution. Subsequently, to compare them precisely, we increased the peak frequency of the Ricker wavelet. We studied the following four cases: 50, 66, 80, and 100 Hz. The maximum frequencies for each case become 125, 165, 200, and 250 Hz. Therefore, 8, 6, 5, and 4 nodal spaces are assured for the shortest wavelength. Figure 7 shows good convergence between the analytical and numerical solutions. It is difficult to rank the methods. Thus, we evaluated the error value *E* as

$$E = \sum_{j=n_a}^{n_b} \left(s_j - s_j^a\right)^2 \div \sum_{j=n_a}^{n_b} \left(s_j^a\right)^2 , \qquad (6.1)$$

where n_a and n_b are the numbers of the start and end sampling time steps, respectively; s_j is the numerical value of the particular seismogram at sample *j*; and $s^a{}_j$ is the corresponding analytical value. The *E* values for each seismogram are displayed in Figure 7, although even after evaluating them, it is still hard to say which is better. Finally, in Figure 8 we plot the seismogram of the specific time window between 0.2 and 0.3 s in Figure 7. Since the factor defining the stability condition is the medium with the minimum wave propagation velocity, the error term mostly appears in the PS reflected wave. We can clearly recognize grid dispersion in the resolution of the 100 Hz peak source case. Remarkably, small grid dispersion can be seen in the result for the DEFGM for the 80 Hz peak source case. On the other hand, the *z*-component of the FDM4 resolution shows a faster approach than the exact waveform. It is difficult to distinguish between the two methods in terms of accuracy. We conclude that FDM4 performs better than the DEFGM since the computational cost of FDM4 is lower than for the DEFGM. Grid dispersion occurs when the S-wave propagates in the upper layer for both cases. However, the reflection ratio is accurately simulated.

7. Rayleigh wave

Lamb’s problem is suitable for evaluating the newly proposed numerical simulation scheme since the analytical solution with a flat free surface is known. Here we used the analytical solution from Saito (1993). The model discussed in this section is a homogeneous half-space medium, which is defined by Vp = 1732 m/s, Vs = 1000 m/s, and ρ = 1500 kg/m^3. The total calculation area is 4001 × 2001 nodes with no absorbing boundary condition. The nodal spacing is set at $\Delta x = \Delta z = 1$ m. The input waveform is a 50-Hz peak Ricker wavelet that acts as a vertical line stress to the surface. Among the 4001 × 2001 nodes, the source point is placed at (1001, 1), and the five receiver points are at (1101, 1), (1201, 1), (1501, 1), (2001, 1), and (3001, 1). Therefore, the receivers are set at 100, 200, 500, 1000, and 2000 m from the source point. The maximum amplitude of the Ricker wavelet is $F_z = 1$ N/m, and the sampling interval is 0.1 ms. The total number of samples in time is 30 000 steps. Since 125 Hz is the approximate maximum frequency of the adopted Ricker wavelet, the shortest S-wavelength becomes 8 m. Thus, 8 nodal spaces are assured for the wavelength. For simplicity, the Rayleigh wave velocity is not taken into this dispersion consideration.

Numerical simulation results

Figure 9a-d shows the waveforms of the particle velocity at the four receiver points. In each subplot, the left and right columns correspond to the horizontal (x) and vertical (z) components, respectively, and we plot three waveforms: The top is the DEFGM solution, the middle is the FDM4 solution, and the bottom is the FDM4 solution under the grid spacing condition $\Delta x = \Delta z = 0.5$ m. The thick black lines and the thin gray lines correspond to the analytic and numerical solutions, respectively. The error values calculated from equation (6.1) are shown in Figure 9. From the figure, we can see that the accuracy of the DEFGM resolution is much better than that of FDM4 even if the grid spacing is set to 0.5 m.

Weight function comparison

In the Rayleigh wave test, we compared results obtained from the following four cases.

- Case 1: 6 bases; this case has been used frequently in recent studies of fracture mechanics (e.g., Beissel and Belytschko, 1996; Liu, 2003; Brighenti, 2005). Thus $\mathbf{P}^{\mathrm{T}}=[1,x,z,xz,x^2,z^2]$ and equation (2.2) with n = 4 are used.
- Case 2: 7 bases; this case has been proposed in this paper. Thus, equations (2.1) and (2.2) with n = 6 are used.
- Case 3: FEM interpolation; this case adopts the full 9 bases, $\mathbf{P}^{\mathrm{T}}=[1,x,z,xz,x^2,z^2,x^2z,xz^2,x^2z^2]$. This case is the same as standard FEM because the term $\left[B^{\mathrm{T}}W_iB\right]^{-1}B^{\mathrm{T}}W_i$ in equation (2.9) becomes B^{-1} for any weight function W_i.
- Case 4: Compound bases; this case adopts $\mathbf{P}^{\mathrm{T}}=[1,x,z,xz,x^2,z^2,x^2z^2]$ and n = 5 only for the central GL point (i = iv in Figure 1). For the other eight GL points, the full 9 bases $\mathbf{P}^{\mathrm{T}}=[1,x,z,xz,x^2,z^2,x^2z,xz^2,x^2z^2]$ are used.

Figure 10(a) and (b) show the results for the above four cases at offsets of 1000 and 2000 m, respectively. The 6-bases case shows large grid dispersion and the error is larger than for FDM4. This case does not work effectively for our DEFGM. When the x^2z^2 term is added to the base vector and the n value is changed to 6 (Case 2), the accuracy is dramatically improved. Among the four cases, Case 4 with compound bases shows the best convergence.

Belytschko et al. (1994) applied the EFGM to an elastic body deformation problem and showed that the EFGM performs better than the standard FEM. However, this good performance is seen only when every element is structured by 2 × 2 nodal points and 2 × 2 GL points. Figure 11 shows a schematic of the EFGM applied to a wave propagation problem. If we use first-order polynomial interpolation (second-order spatial accuracy) as shown in Figure 11(a), there is a high probability that MLS interpolation on the GL points will improve the linearly interpolated waveform in the standard FEM. If we use second-order polynomial interpolation (third-order spatial accuracy) as shown in Figure 11(b), the standard FEM produces a far better interpolated waveform than does the first-order case. The scope for MLS interpolation to increase accuracy becomes smaller than when we use first-order polynomial interpolation. In fact, even the 7-base case proposed in this paper performs with less accuracy than the standard FEM. On the other hand, the compound-base case performs with better accuracy than the standard FEM. We think the reason for this differing performance is as follows. On the GL points near the boundary, the continuation of the shape function between neighboring elements is more important than the improvement of the shape function by MLS interpolation. On the GL point at the center of the element, MLS interpolation effectively improves the shape function.

8. Discussion and Conclusion

We have proposed a numerical method called the decomposed element-free Galerkin method (DEFGM). We examined our scheme by using numerical simulations of PS reflection and Lamb's problems.

The DEFGM decomposes a stiffness matrix in the element-free Galerkin method into individual schemes and adapts an explicit time-update scheme. In other words, the DEFGM solves elastic wave equation problems by alternatively updating stress-strain relations and equations of motion as in the staggered-grid finite difference method (FDM). To examine this idea, we used the DEFGM with 4001 × 2001 nodal points, which was sufficient to compare the performances of the DEFGM and the finite difference method with fourth-order spatial accuracy (FDM4). We also compared the performance of the DEFGM with FDM4 in terms of computation time and numerical accuracy. The DEFGM required a CPU time that was at least 2.9 times that of FDM4. When we used PS reflection waves for the comparison, we found that the numerical accuracy of FDM4 was a little better than that of the DEFGM. This is because the spatial accuracy of FDM4 is fourth-order while that of the DEFGM we used is third-order. Therefore, FDM4 is better than the DEFGM when infinite space is simulated since FDM4 consumes less CPU time than the DEFGM. However, when we simulated solutions to Lamb's problem with 8 nodal spaces for the shortest S-wavelength, the DEFGM provided an accurate Rayleigh waveform for a distance at least equal to 50 wavelengths, while for FDM4 it was 5 wavelengths. This is because the FDM4 we used adopts second-order spatial accuracy near the free surface (Figure 5). In addition, the DEFGM with a 1-m nodal spacing was more accurate than FDM4 with 0.5-m grid spacing. In this comparison, the CPU time of the DEFGM was less than that of FDM4. Finally, we compared the results from some weight functions. Although the weight function used by Beissel and Belytschko (1996), Liu (2003), and Brighenti (2005) was less accurate than expected, the proposed combination of base vector and weight function dramatically improved the accuracy of the EFGM. However, we found a specific DEFGM (compound-base case) that had better accuracy than the standard FEM.

In summary, the numerical performance of the EFGM is clearly improved by the DEFGM. This is because the DEFGM requires a computation memory size comparable to the FDM. Moreover, the DEFGM can accommodate perfectly matched layer (PML) absorbing boundary conditions as in the FDM case. A comparison using an exact analytical solution of PS reflection waves showed that the results of the DEFGM are as accurate as those of FDM4. The CPU time of FDM4 was smaller than that of the DEFGM. However, a comparison using Lamb's problem showed that the DEFGM provides ten or more times better resolution than FDM4. If we obtain accurate DEFGM resolution by using FDM4, the CPU time of the DEFGM becomes smaller than for FDM4. In addition, we found a specific DEFGM that performs with better accuracy than the standard FEM. The DEFGM is thus able to improve the shape function of the standard FEM. The DEFGM could be suitable for numerical simulations of elastic wavefields, especially in cases where a free surface is considered.

Because the weight functions used in this study were only experimental, it is possible that the accuracy of the DEFGM could be increased. Future studies should investigate which weight functions best approximate the wave equations and apply them to realistic 3D problems.

Acknowledgments

We appreciate the advice and encouragement of Jun Ichi Takekawa of the Geo-Research Institute and the technical support of Norihiro Nakata of Kyoto

University. We gratefully acknowledge the anonymous reviewers who helped us improve the manuscript.

Appendix A. Perfectly matched layer

Because the DEFGM is developed by using the stress-velocity formulation, PML absorbing boundary condition can be applied as shown in Collino and Tsogka (2001). Equations (2.19a–c) are divided as follows:

$$\frac{\partial \sigma_{xx}^{x}}{\partial t}+d^{x'}\sigma_{xx}^{x}=\left(\lambda+2\mu\right)\frac{\partial v_{x}}{\partial x}, \qquad (A.1a)$$

$$\frac{\partial \sigma_{xx}^{z}}{\partial t}+d^{z'}\sigma_{xx}^{z}=\lambda\frac{\partial v_{z}}{\partial z}, \qquad (A.1b)$$

$$\frac{\partial \sigma_{zz}^{x}}{\partial t}+d^{x'}\sigma_{zz}^{x}=\lambda\frac{\partial v_{x}}{\partial x}, \qquad (A.1c)$$

$$\frac{\partial \sigma_{zz}^{z}}{\partial t}+d^{z'}\sigma_{zz}^{z}=\left(\lambda+2\mu\right)\frac{\partial v_{z}}{\partial z}, \qquad (A.1d)$$

$$\frac{\partial \sigma_{xz}^{x}}{\partial t}+d^{x'}\sigma_{xz}^{x}=\mu\frac{\partial v_{z}}{\partial x}, \qquad (A.1e)$$

$$\frac{\partial \sigma_{xz}^{z}}{\partial t}+d^{z'}\sigma_{xz}^{z}=\mu\frac{\partial v_{x}}{\partial z}, \qquad (A.1f)$$

where

$$\sigma_{kl}=\sigma_{kl}^{x}+\sigma_{kl}^{z}, \quad (k,\, l=x,\, z). \quad (A.2)$$

This division is called directional splitting. The two-dimensional PML therefore has twice the memory cost of the non-PML case. The damping functions are given as follows:

$$d^{k'}=\frac{3V_p}{2D}\log\left(\frac{1}{R}\right)\left(\frac{k'}{D}\right)^2 \quad (k=x,\,z), \quad \text{(A.3)}$$

where *k′* is the distance from the boundary between the PML and non-PML area (see Figure 12), *D* is the width of the PML, V_p is the P-wave velocity, and *R* is the theoretical reflection coefficient after discretization, which can be chosen to be a very small number (e.g., 0.001, which is the value used in this paper). For numerical computations, the term $d\sigma$ is evaluated by the average value between the old and new times, thus

$$d\sigma=d\frac{\sigma^{t+\Delta t/2}+\sigma^{t-\Delta t/2}}{2}. \qquad \text{(A.4)}$$

Equations of motion also adopt the PML; equation (3.1) becomes

$$\rho\left(\frac{\partial v_k^l}{\partial t}+d^{l'}v_k^l\right)=f_k^l+\frac{\partial\sigma_{kl}}{\partial l},\ (k,\,l=x,\,z)\ , \text{(A.5)}$$

where

$$v_k=v_k^x+v_k^z,\ (k=x,\,z), \qquad \text{(A.6)}$$

$$f_k=f_k^x+f_k^z,\ (k=x,\,z). \qquad \text{(A.7)}$$

The treatment for *dv* is the same as for equation (A.4).

References

Beissel, S. and Belytschko, T. (1996). Nodal integration of the element-free Galerkin method, *Comput. Methods Appli. Mech. Engng.* **139**, 49–74.

Belytschko, T., Lu, Y. Y. and Gu. L. (1994). Element-free Galerkin method, *Int. J. Numer. Meth. Engng.*, **37**, 229–256.

Bohlen, T. and Saenger, E. H. (2006). Accuracy of heterogeneous staggered-grid finite-difference modeling of Rayleigh waves, *Geophysics,* **71**, T109–T115.

Brighenti, R. (2005). Application of the element-free Galerkin meshless method to 3D fracture mechanics problems, *Engng. Fracture Mech.,* **72**, 2808–2820.

Collino, F. and Tsogka, C. (2001). Application of the perfectly matched absorbing layer model to the linear elastodynamic problem in anisotropic heterogeneous media, *Geophysics,* **66**, 294–307.

Dumbser, M. and Käser, M. (2006). An arbitrary high order discontinuous Galerkin method for elastic waves on Unstructured meshes II: The three-dimensional isotropic case, *Geophys. J. Int.*, **167(1)**, 319–336.

Graves, R. W. (1996). Simulating seismic wave propagation in 3D elastic media using staggered-grid finite differences, *Bull. Seims. Soc. Am.*, **86**, 1091–1106.

Ichimura, T., Hori, M. and Kuwamoto, H (2007). Earthquake motion simulation with multiscale finite-element analysis on hybrid grid, *Bull. Seims. Soc. Am.*, **97**, 1133–1143.

Jia, X. and Hu, T. (2006). Element-free precise integration method and its applications in seismic modeling and imaging, *Geophys. J. Int.*, **166**, 349–372.

Käser, M. and Dumbser, M. (2006). An arbitrary high order discontinuous Galerkin method for elastic waves on Unstructured meshes I: The two-dimensional isotropic case with external source terms, *Geophys. J. Int.*, **166(2)**, 855–877.

Koketsu, K., Fujiwara, H. and Ikegami, Y. (2004) Finite-element simulation of seismic ground motion with a voxel mesh, *Pure Appl. Geophys.* **161**, 2183-2198.

Komatitsch, D. and Tromp, J. (1999). Introduction to the spectral element method for three-dimensional seismic wave propagation, *Geophys. J. Int.*, **139**, 806–822.

Komatitsch, D. and Tromp, J. (2003). A perfectly matched layer absorbing boundary condition for the second-order seismic wave equation, *Geophys. J. Int.*, **154**, 146–153.

Levander, A. R. (1988). Fourth-order finite-difference P-sv seismograms, *Geophysics*, **53**, 1425–1436.

Liu, G. R. (2003). Mesh free methods, *CRC press.*

Lu, Y. Y., Belytschko, T. and Tabbara, M. (1995). Element-free Galerkin method for wave propagation and dynamic fracture, *Comput. Methods Appl. Mech. Engrg.*, **126**, 131–153.

Ma, S., Archuleta, R. J. and Liu, P. (2004). Hybrid modeling of elastic *P-SV* wave motion: A combined finite-element and staggered-grid finite-difference approach, *Bull. Seims. Soc. Am.*, **94**, 1557–1563.

Min, D. J., Shin, C., Pratt, R. G. and Yoo, H. S. (2003). Weighted-averaging finite-element method for 2D elastic wave equations in the frequency domain, *Bull. Seims. Soc. Am.*, **93**, 904–921.

Saito, M. (1993). Branch line contribution in Lamb's Problem, *Butsuri-Tansa*, **46**, 372–380 (in Japanese).

Spice homepage, http://www.spice-rtn.org.

Virieux, J. (1986). *P-sv* wave propagation in heterogeneous media: Velocity-stress finite difference method, *Geophysics*, **51**, 889–901.

Table 1: The array sizes for the DEFGM and FDM4 for the case in which (2nx+1) × (2nz+1) nodal points are evaluated.

	λ, μ	ρ	$\bar{M}$	σ_{xx}, σ_{zz}, σ_{xz}	v_x, v_z
DEFGM	nx × nz	N/A	(2nx+1) × (2nz+1)	nx × nz × 9	(2nx+1) × (2nz+1)
FDM4	(2nx+1) × (2nz+1)	(2nx+1) × (2nz+1)	N/A	(2nx+1) × (2nz+1)	(2nx+1) × (2nz+1)

Table 2: Summary of calculation times. The calculation model consists of 1000 time steps and 401 × 401 nodal points (upper table). The model with 2000 time steps and 801 × 801 nodal points (lower table) is also shown. The numbers in square brackets are the ratios of the calculation times when the time for FDM4 is assumed to have the value of one.

401 × 401 grids, 1000 steps	**Absorbing B. C.: N/A**	**Absorbing B. C.: PML**
FDM4	**26 s [1]**	**2 min 30 s [5.8]**
DEFGM	**1 min 16 s [2.9]**	**18 min 26 s [43]**

801 × 801 grids, 2000 steps	**Absorbing B. C.: N/A**	**Absorbing B. C.: PML**
FDM4	**4 min 34 s [1]**	**30 min 50 s [6.8]**
DEFGM	**24 min 18 s [5.3]**	**3 h 5 min 34 s [41]**

Figure captions

Figure 1: The rectangular element and grid arrangement for the EFGM scheme for the case of a second-order base function and 3 × 3 Gauss-Legendre integral points. (x_0 , z_0) is the central position of the element; Δx and Δz are the nodal spacings for x- and z-direction, respectively; and G = 0.7745867.

Figure 2: Weight function (equation (2.2)) with n = 6.

Figure 3: Elastic body consisting of nine elements. Open circles are nodal points.

Figure 4: Flow chart of the DEFGM.

Figure 5: Grid arrangement for the staggered-grid finite difference scheme. Δx and Δz are the grid spacings for the x- and z-directions, respectively. We chose a free surface boundary by the Levander (1988) method.

Figure 6: Upper left: the calculation model. Upper right: a snapshot of the z-component

of the particle velocity at 0.10 s. Lower left and right: *z*-components at 0.14 and 0.18 s, respectively.

Figure 7: Comparison of the x- and z-direction velocity components. The analytical solution (thick black line) is plotted against the numerical one (thin gray line) obtained by the DEFGM and FDM4.

Figure 8: Magnification of Figure 7 between 0.2 and 0.3 s. The analytical solution (thick black line) is plotted against the numerical one (thin gray line) obtained by the DEFGM and FDM4.

Figure 9: Comparison of the *x*- and *z*-direction velocity components. The analytical solution (thick black line) is plotted against the numerical one (thin gray line) obtained by the DEFGM and FDM4. From top to bottom, the graphs correspond to the DEFGM, FDM4 with 1-m grid spacing, and FDM4 with 0.5-m grid spacing. Offsets are (a) 100 m, (b) 200 m, (c) 500 m, and (d) 1000 m.

Figure 10: Comparison of the *x*- and *z*-direction velocity components. The analytical

solution (thick black line) is plotted against the numerical one (thin gray line) obtained by various DEFGMs. From top to bottom, the graphs correspond to 6 bases, 7 bases, FEM interpolation, and compound bases. Offsets are (a) 1000 m and (b) 2000 m.

Figure 11: Schematics of the EFGM computation as conducted for a wave propagation problem. (a) First-order polynomial interpolation, (b) Second-order polynomial interpolation.

Figure 12: The PML damping function (A.3).

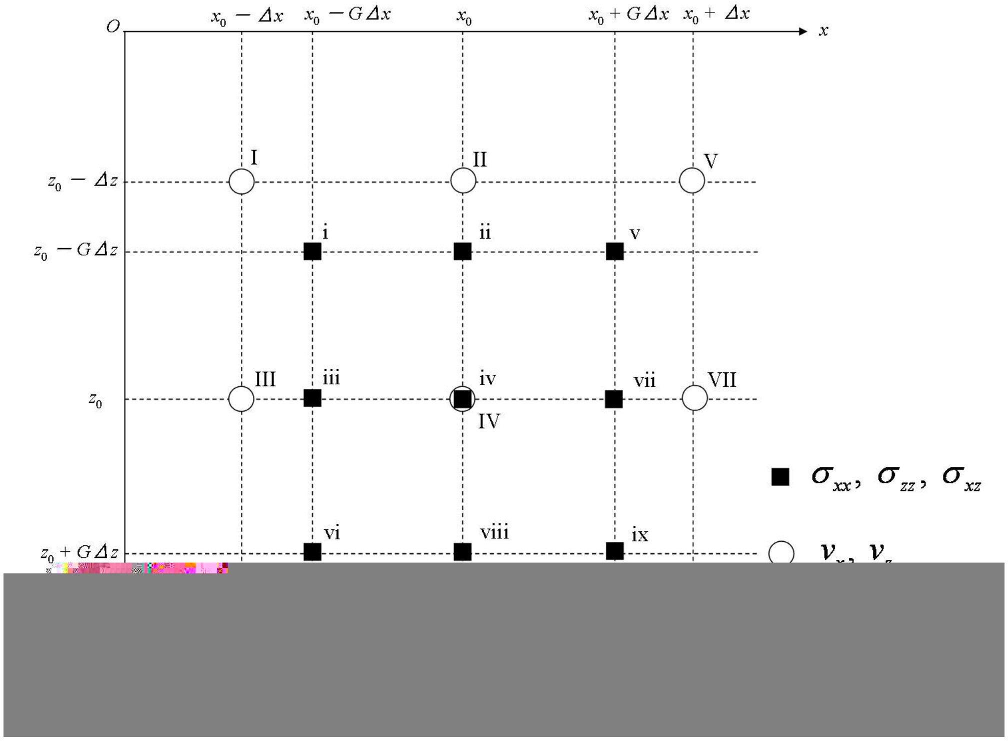

Fig.1: The rectangular element and grid arrangement for the EFGM scheme for the case of a second-order base function and 3 X 3 Gauss-Legendre integral points. (x0 , z0) is the central position of the element; Δx and Δz are the nodal spacings for x- and z-direction, respectively; and G = 0.7745867.

254x190mm (150 x 150 DPI)

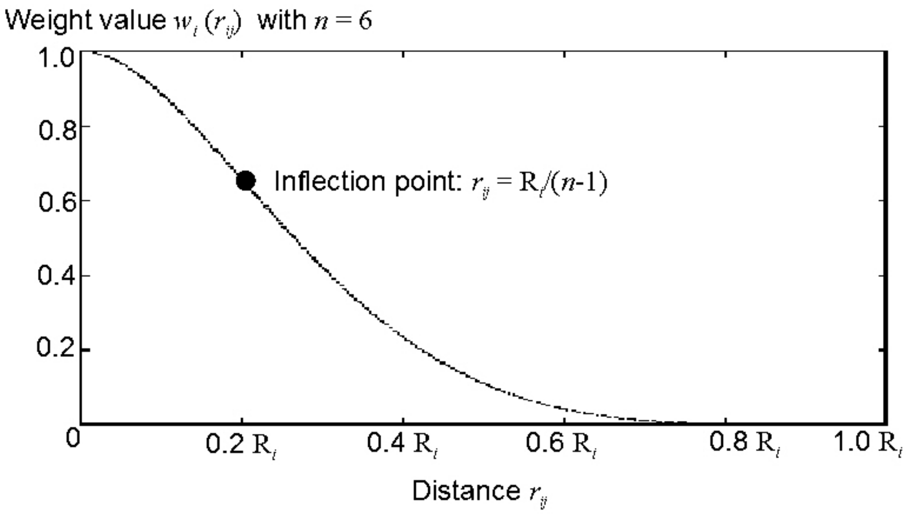

Fig.2: Weight function (equation (2.2)) with n = 6.

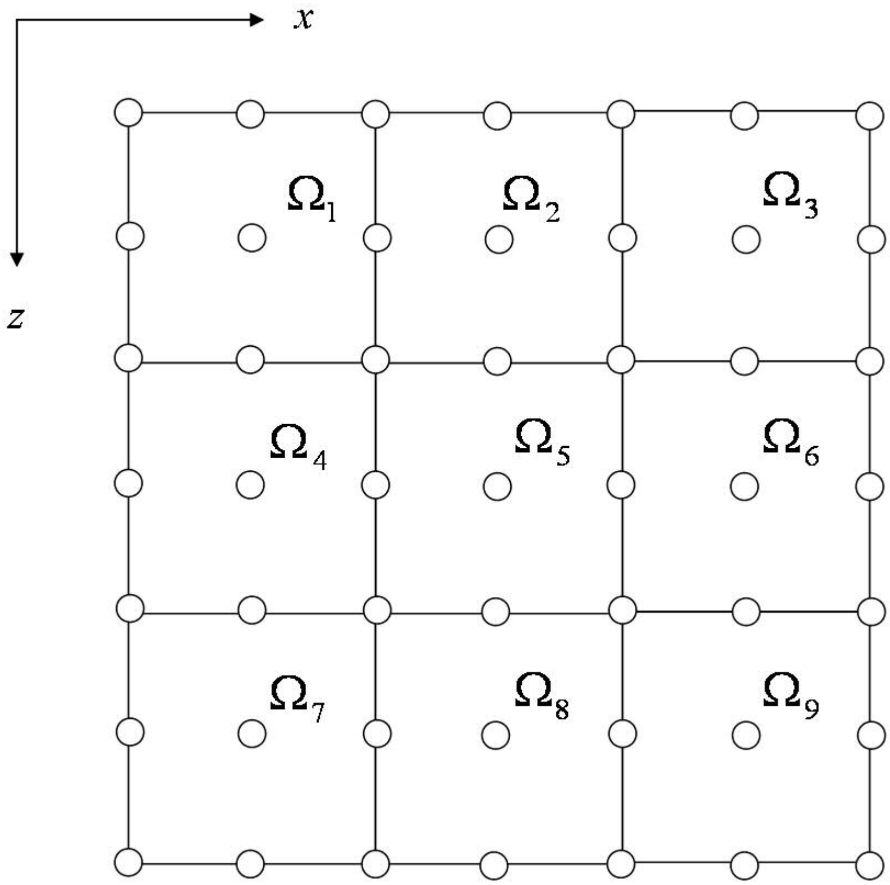

Fig.3: Elastic body consisting of nine elements. Open circles are nodal points.
154x145mm (150 x 150 DPI)

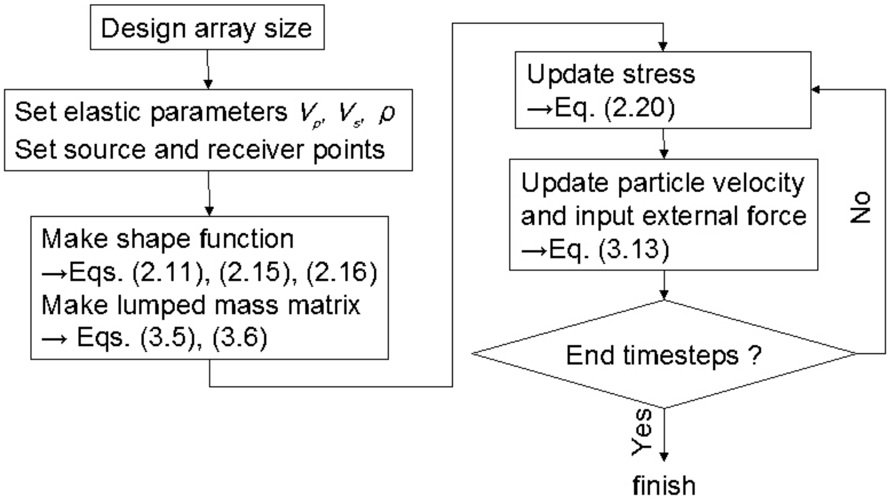

Fig.4: Flow chart of the DEFGM.

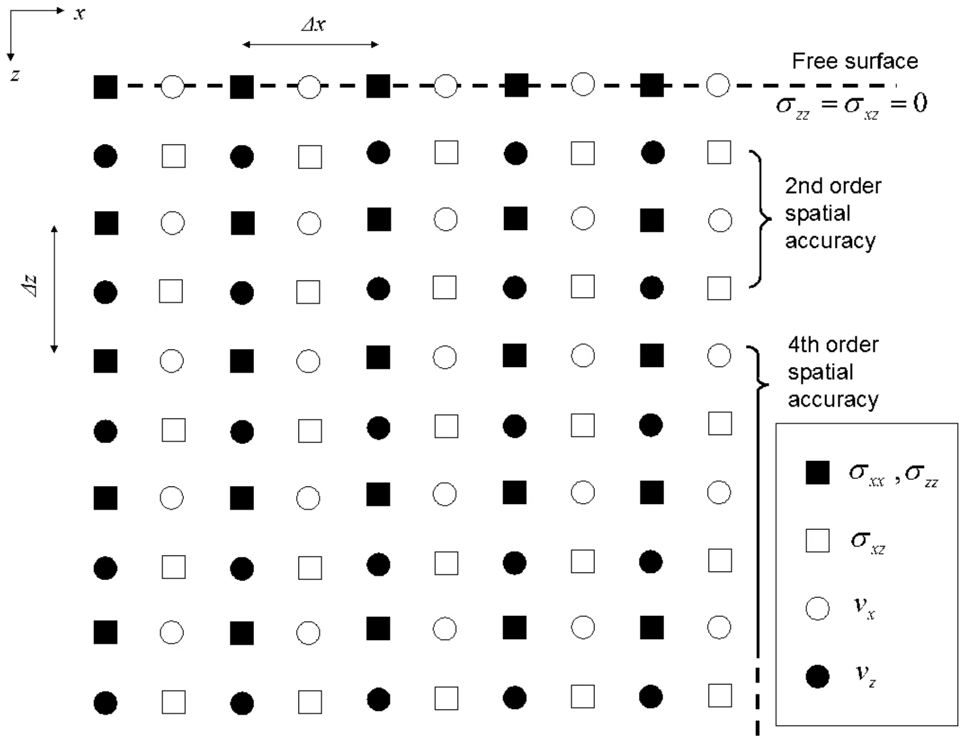
x
z
Δx
Δz
Free surface
$\sigma_{zz} = \sigma_{xz} = 0$
2nd order spatial accuracy
4th order spatial accuracy
σ_{xx}, σ_{zz}
σ_{xz}
v_x
v_z

Fig.5: Grid arrangement for the staggered-grid finite difference scheme. Δx and Δz are the grid spacings for the x- and z-directions, respectively. We chose a free surface boundary by the Levander (1988) method.

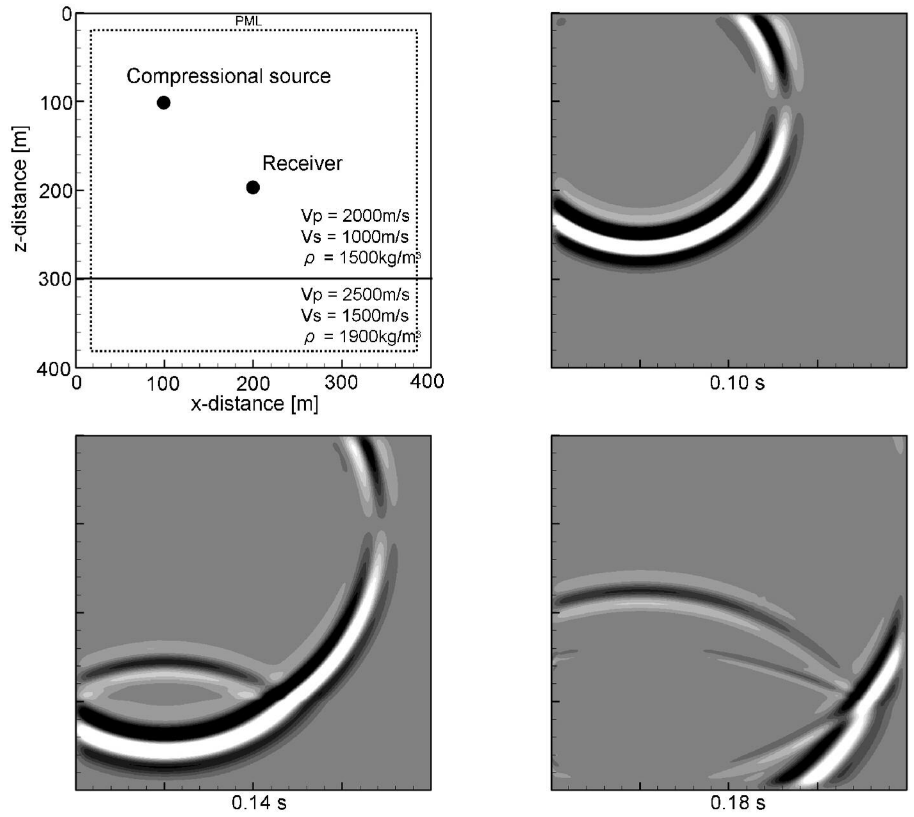

Fig.6: Upper left: the calculation model. Upper right: a snapshot of the z-component of the particle velocity at 0.10 s. Lower left and right: z-components at 0.14 and 0.18 s, respectively.

480x421mm (150 x 150 DPI)

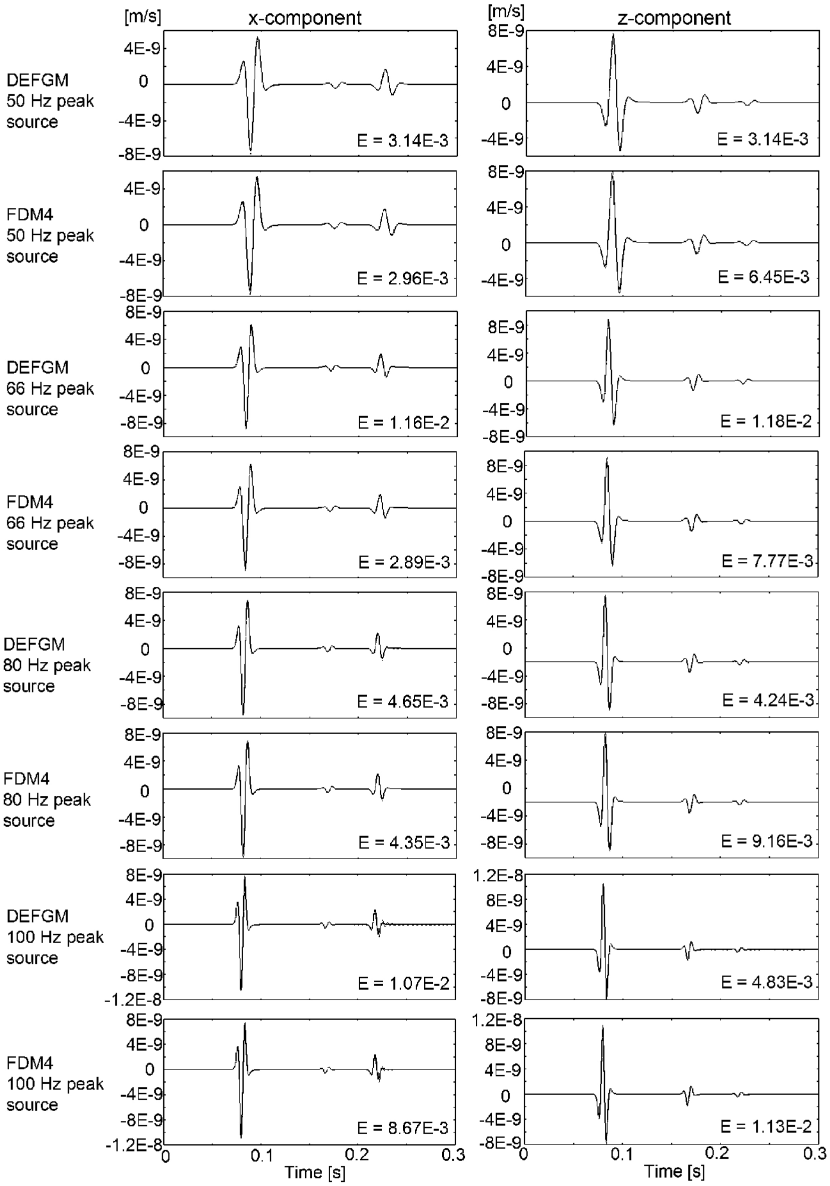

Fig.7: Comparison of the x- and z-direction velocity components. The analytical solution (thick black line) is plotted against the numerical one (thin gray line) obtained by the DEFGM and FDM4.

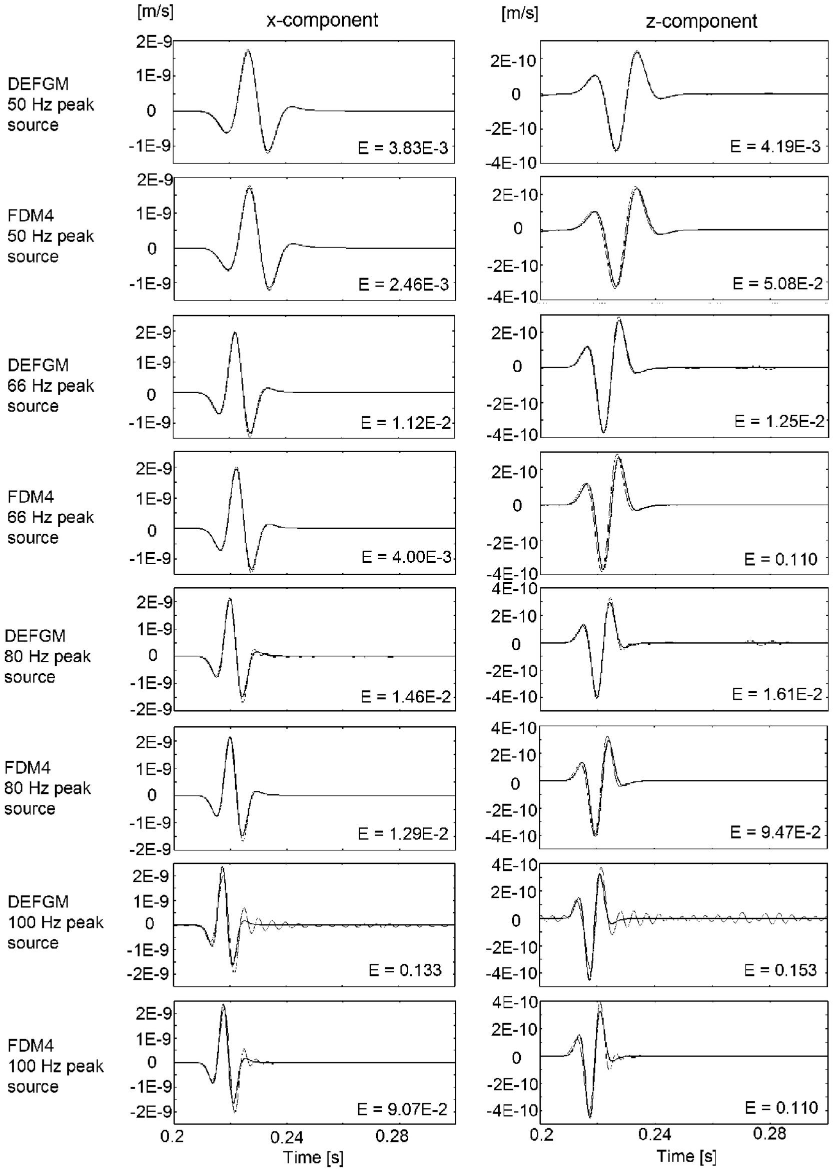

Fig.8: Magnification of Figure 7 between 0.2 and 0.3 s. The analytical solution (thick black line) is plotted against the numerical one (thin gray line) obtained by the DEFGM and FDM4.

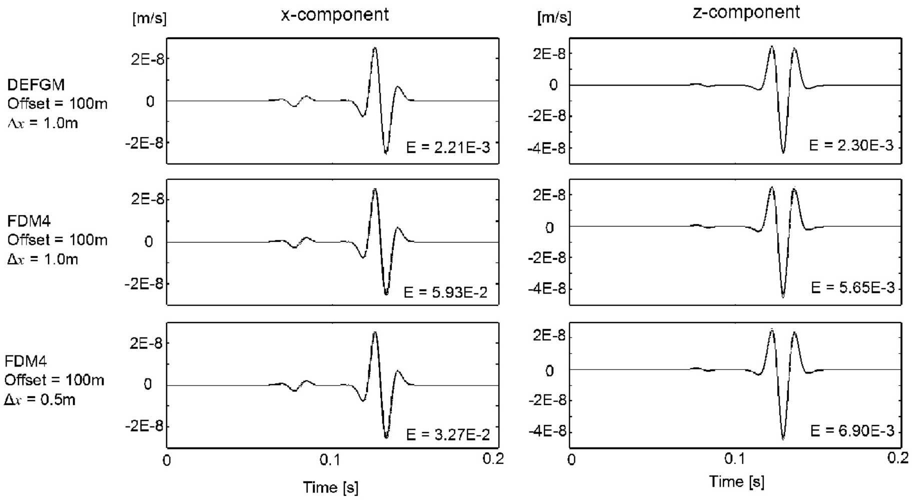

Fig.9(a): Comparison of the x- and z-direction velocity components. The analytical solution (thick black line) is plotted against the numerical one (thin gray line) obtained by the DEFGM and FDM4. From top to bottom, the graphs correspond to the DEFGM, FDM4 with 1-m grid spacing, and FDM4 with 0.5-m grid spacing. Offsets are 100 m

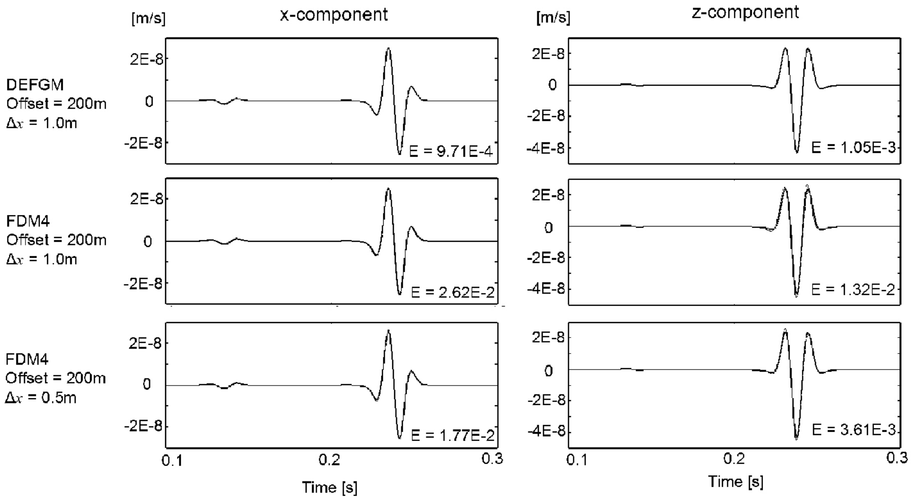

Fig9(b): offset = 200 m

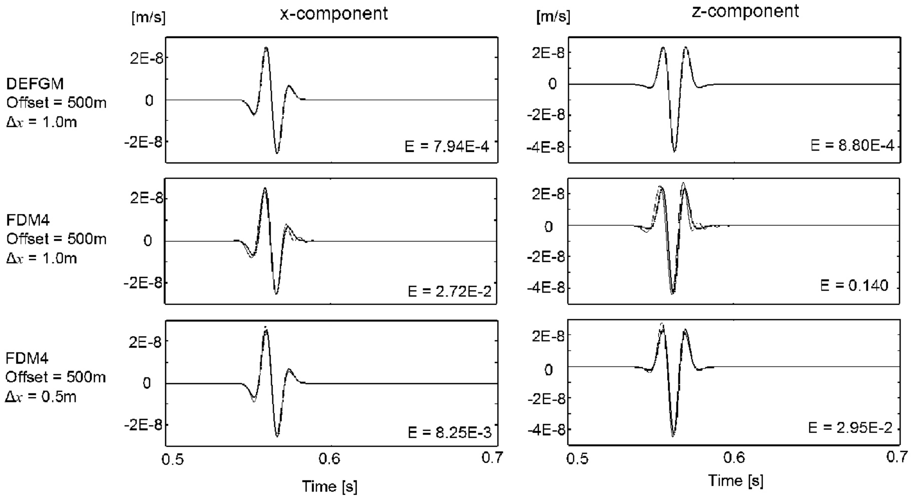

Fig9(c): offset = 500 m

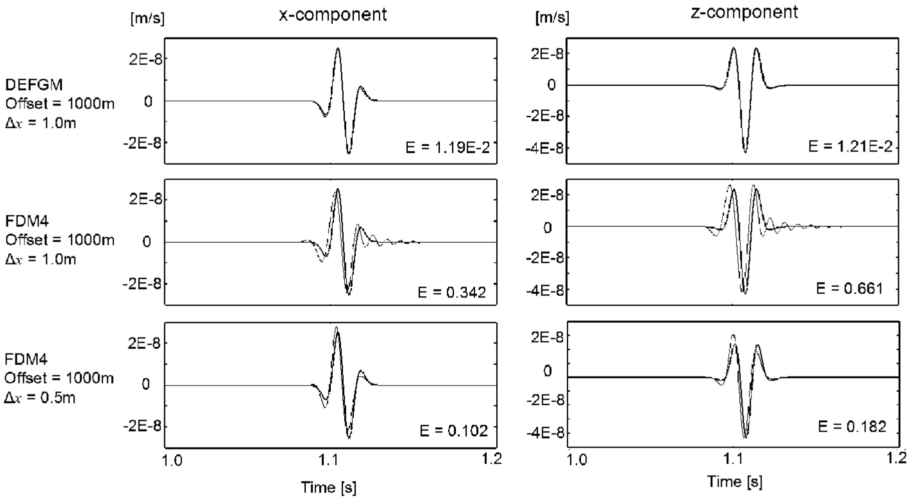

Fig9(d): offset = 1000 m

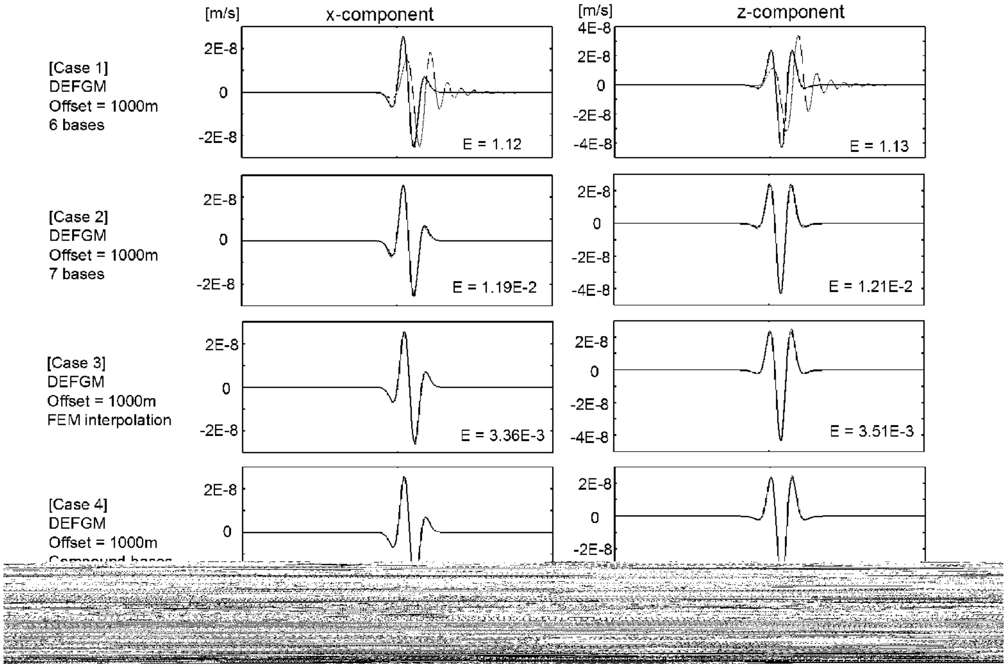

Fig.10(a): Comparison of the x- and z-direction velocity components. The analytical solution (thick black line) is plotted against the numerical one (thin gray line) obtained by various DEFGMs. From top to bottom, the graphs correspond to 6 bases, 7 bases, FEM interpolation, and compound bases. Offsets are 1000 m

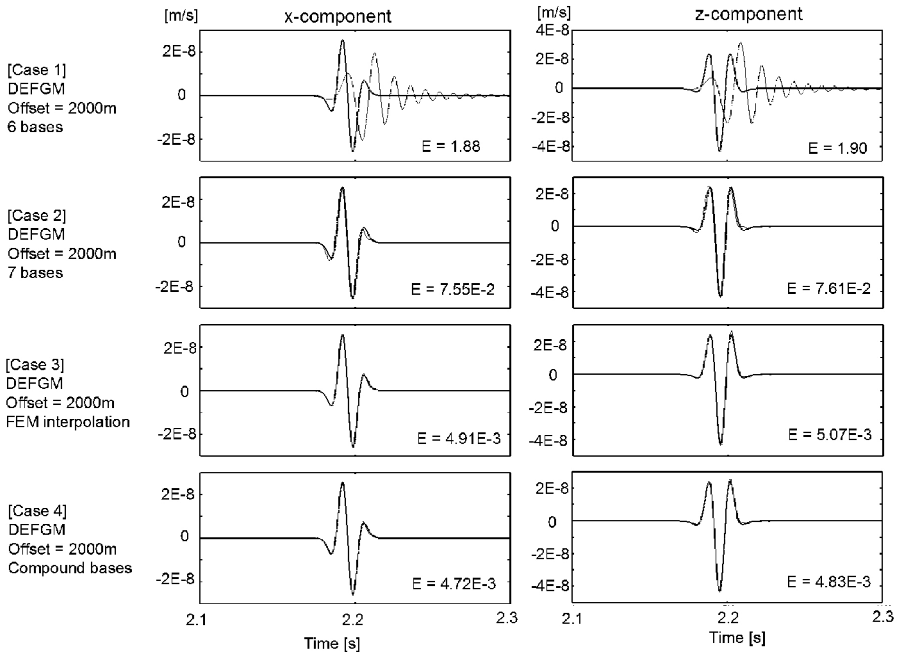

Fig.10(b): offset = 2000 m

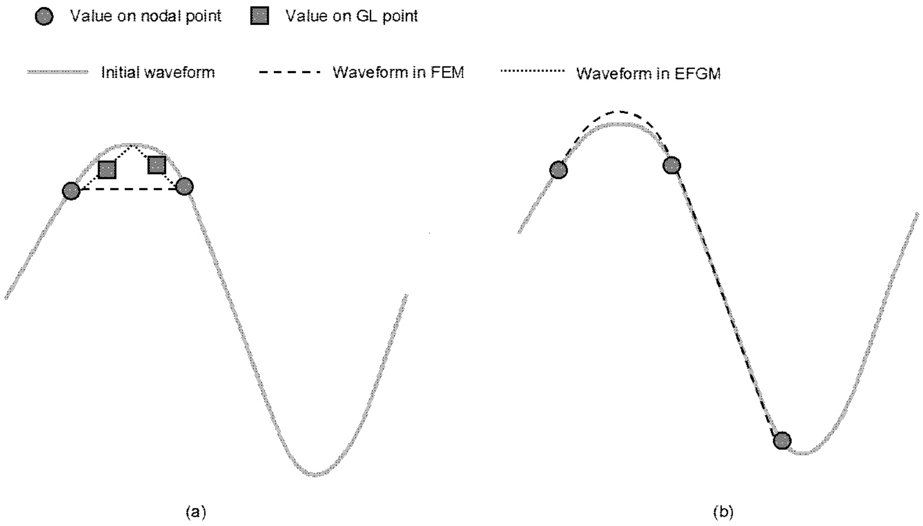

Fig.11: Schematics of the EFGM computation as conducted for a wave propagation problem. (a) First-order polynomial interpolation, (b) Second-order polynomial interpolation.

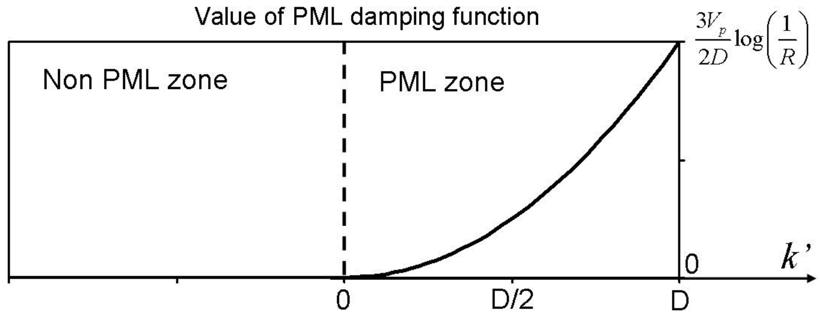

Fig.12: The PML damping function (A.3).
226x105mm (150 x 150 DPI)