A support-operator method for viscoelastic wave modelling in 3-D heterogeneous media

Geoffrey P. Ely,[1] Steven M. Day[2] and Jean-Bernard Minster[1]

[1]*Scripps Institution of Oceanography, University of California, San Diego,* 9500 *Gilman Drive, La Jolla,* CA 92093-0225, *USA. E-mail: gely@ucsd.edu*
[2]*Department of Geological Sciences, San Diego State University,* 5500 *Campanile Drive, San Diego,* CA 92182-1020, *USA*

SUMMARY
We apply the method of support operators (SOM) to solve the 3-D, viscoelastic equations of motion for use in earthquake simulations. SOM is a generalized finite-difference method that can utilize meshes of arbitrary structure and incorporate irregular geometry. Our implementation uses a 3-D, logically rectangular, hexahedral mesh. Calculations are second-order in space and time. A correction term is employed for suppression of spurious zero-energy modes (hourglass oscillations). We develop a free surface boundary condition, and an absorbing boundary condition using the method of perfectly matched layers (PML). Numerical tests using a layered material model in a highly deformed mesh show good agreement with the frequency-wavenumber method, for resolutions greater than 10 nodes per wavelength. We also test a vertically incident *P* wave on a semi-circular canyon, for which results match boundary integral solutions at resolutions greater that 20 nodes per wavelength. We also demonstrate excellent parallel scalability of our code.

Key words: Numerical solutions; Earthquake ground motions; Computational seismology; Wave propagation.

GJI Seismology

1 INTRODUCTION

The finite-difference method (FDM) has been extensively used for modelling 3-D seismic wave propagation and rupture-dynamics problems. The fourth-order staggered-grid scheme, nicely summarized by Graves (1996), is particularly well suited for large-scale, high-resolution problems because it is accurate, efficient and readily parallelized for multiprocessor execution. Recent applications include basin wave propagation by Olsen *et al.* (2006) and rupture dynamics by Dalguer & Day (2007).

Many such FDM implementations, however, are restricted to rectilinear discretizations of the problem domain, which in some cases leads to a poor approximation of problem geometry. Surface topography, for example, may be neglected. Ma *et al.* (2007) show with finite-element simulations that the topography of the San Bernardino Mountains may disrupt surface waves generated on the adjacent San Andreas Fault, effectively shielding Los Angeles. The restriction to rectilinear meshes also represents a severe constraint on dynamic rupture simulations, usually restricting consideration to planar, vertical faults that coincide with grid planes. This constraint can be relaxed, as in Cruz-Atienza (2006), for example, but at a substantial sacrifice of accuracy. Alternatively, a coordinate mapping can be introduced to conform the FDM grid to a non-planar fault geometry, as in Kase & Day (2006), for example.

A variety of approaches have been taken to enable non-rectangular geometry in earthquake simulations. To cite a few examples, Aagaard (1999) used the finite-element method (FEM) on tetrahedral meshes; Oglesby (1999), and Oglesby *et al.* (2000) employed FEM on hexahedral meshes; Komatitsch & Tromp (1999) used the spectral element method and Dumbser & Kašer (2006) used a high-order discontinuous Galerkin method.

We apply the method of support-operators (SOM) developed by Samarskii *et al.* (1981, 1982) and Shashkov (1996). SOM is a general scheme for discretizing the differential form of partial differential equations. Many simple FDMs and FEMs are special cases of SOM. The approach constructs discrete analogues of continuum derivative operators that satisfy important integral identities, such as the adjoint relation between gradient and divergence. SOM brings to an FDM-type formulation the FEM advantage that energy is conserved in the semi-discrete equations (i.e. spatially discrete but before time discretization), as an immediate consequence of the adjoint relations. Likewise, the adjoint relations ensure that seismic reciprocity is satisfied by the fully discrete equations.

SOM may employ structured or unstructured meshes, and be extended to high order of accuracy. We develop the theory for arbitrary order on structured meshes, and implement the second order case for our numerical algorithm. Rojas *et al.* (2007) use a related finite-difference formulation to model earthquake rupture dynamics in 2-D. They implement a fourth-order scheme, but restrict it to Cartesian meshes. Their approach is similar to ours, in that they form the difference operators using the adjoint relations, but differs from our SOM formulation in the nature of the mesh employed (they use a staggered mesh of the form used in, e.g. Levander 1988).

This paper begins with a derivation of 3-D SOM spatial difference operators, followed by a comparison to the related method of one-point quadrature. A full listing of both types of operators is located in the Appendix. We then layout the numerical algorithm for solving wave equations with hourglass corrections, and perfectly matched layer (PML) absorbing boundaries. Presented next are numerical tests using a layered model, that is verified against wavenumber integration solutions, and a semi-circular canyon model that is verified against boundary integral solutions. Finally, we examine the parallel processing capability of our algorithm. In the concluding discussion, we look at potential enhancements to, and applications of our method.

2 THE SUPPORT OPERATOR METHOD

The linearized equations of motion for isotropic viscoelastic motion are

$$g_{ij} = \partial_j (u_i + \gamma v_i), \tag{1}$$

$$\sigma_{ij} = \lambda \, \delta_{ij} \, g_{kk} + \mu (g_{ij} + g_{ji}), \tag{2}$$

$$a_i = \frac{1}{\rho} \, \partial_j \sigma_{ij}, \tag{3}$$

$$\dot{v}_i = a_i, \tag{4}$$

$$\dot{u}_i = v_i, \tag{5}$$

where $\boldsymbol{\sigma}$ is the stress tensor, $\mathbf{u}$ and $\mathbf{v}$ are displacement and velocity vectors, ρ is density, λ and μ are elastic moduli and γ is viscosity. The Kelvin–Voigt model of viscosity used here is not of a realistic form for seismological problems, but is included in the formulation in anticipation of future applications to non-linear problems (e.g. rupture dynamics) where viscous losses may be needed for numerical regularization. Realistic attenuation can be readily added by the memory-variable technique introduced by Day & Minster (1984) and since refined by various authors (Moczo *et al.* 2006, provide a comprehensive review). All variables are functions of position $\mathbf{x}$, while $\boldsymbol{\sigma}$, $\mathbf{u}$ and $\mathbf{v}$ are time dependent as well. It is generally not possible to find analytical solutions to this system of equations, so we must rely on numerical methods to find approximate solutions.

We first discretize the field variables and material constants onto a hexahedral mesh that has a logically rectangular structure. We adapt the general formalism of Shashkov (1996) to the equations of 3-D elastodynamics. Following that general formalism, we define two types of spatial discretizations on the mesh. Nodal functions are defined at the node points. We denote the space of nodal functions by H^N. Cell functions represent average function values over element volumes. We denote the space of cell functions by H^C. Mesh node points are located in the 3-D logical structure by their indices j, k and l, and each interior node is shared by exactly eight elements. For the structured mesh there is a mapping from Cartesian space to logical space, $\mathbf{x} \rightarrow \boldsymbol{\xi}$ (Fig. 1) and the nodal coordinates $\mathbf{X}_{jkl} \in H^N$ map to the logical coordinates $\boldsymbol{\Xi}_{jkl}$.

We will need two types of discrete difference operators, one that operates on nodal functions and one that operates on cell functions. We begin with the derivation of the first one: an operator that computes the derivative of a nodal function with a cell function result. Consider the nodal discretization of an unknown function: $F_{jkl} \in H^N$. Polynomial interpolation of the discrete function F_{jkl} can be used to construct a continuous function $f(\boldsymbol{\xi})$ that is an approxima-
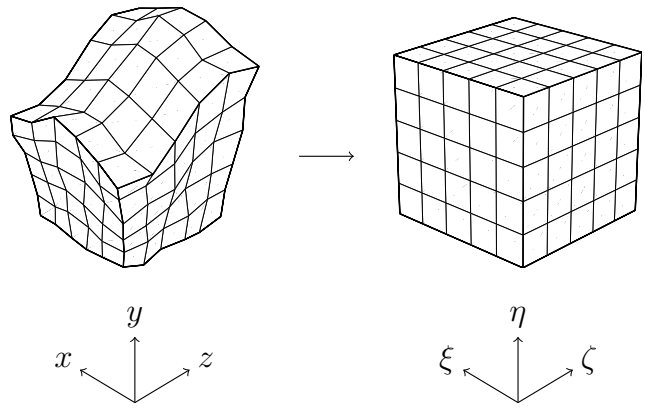

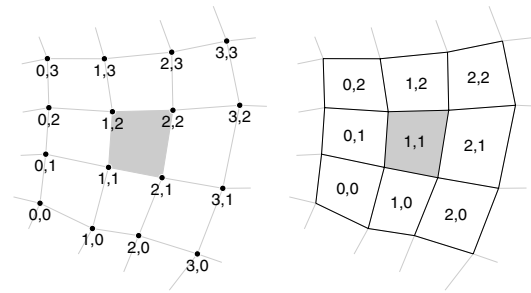**Figure 1.** Map of Cartesian space to logical space.

Figure 2. Stencil node and cell indexing for cubic ($n = 3$) interpolation in the shaded cell (1,1). For simplicity only two dimensions are shown instead of three.

tion to the original unknown function. Provided the original function is smooth and well behaved, the error of approximation depends on the polynomial degree n.

We restrict the interpolation scheme such that, for a particular cell, $f(\boldsymbol{\xi})$ depends only on the nodes in the immediate vicinity. This group of nodes is called the stencil, and is diagrammed in Fig. 2. The stencil nodes have the indices $j, k, l = 0, \ldots, n$ and the stencil cells have the indices $j, k, l = 0, \ldots, n - 1$. The value of n must be odd to insure logical symmetry about the central cell. The index of the central cell is (m, m, m) where $m = (n - 1)/2$.

The interpolation function is given by

$$f(\boldsymbol{\xi}) = \sum_{j,k,l=0}^{n} N_{jkl}(\boldsymbol{\xi}) F_{jkl}, \tag{6}$$

where the shape functions $N_{jkl}(\boldsymbol{\xi})$ are formed from Lagrange interpolation polynomials

$$N_{jkl}(\boldsymbol{\xi}) = \ell_j^n(\xi) \ell_k^n(\eta) \ell_l^n(\zeta), \tag{7}$$

and the Lagrange polynomials of degree n are given by

$$\ell_j^n(\xi) = \prod_{\substack{i=0 \\ i \neq j}}^{n} \frac{\xi - \Xi_i}{\Xi_j - \Xi_i}. \tag{8}$$

Evaluating the shape functions at the nodes gives

$$N_{jkl}(\Xi_{pqr}) = \delta_{jp} \delta_{kq} \delta_{lr}, \tag{9}$$

so evaluating the interpolation function at the mesh nodes returns exactly the discrete function values

$$f(\Xi_{jkl}) = F_{jkl}. \tag{10}$$

Repeating the interpolation procedure for the nodal coordinates $\mathbf{X}_{jkl}$ results in a mapping from logical coordinates to Cartesian coordinates,

$$\mathbf{x}(\boldsymbol{\xi}) = \sum_{j,k,l=0}^{n} N_{jkl}(\boldsymbol{\xi})\mathbf{X}_{jkl}. \tag{11}$$

We can use this mapping to find the gradient of f by solving the system of equations

$$\frac{\partial f}{\partial \boldsymbol{\xi}} = (\nabla f) \cdot \frac{\partial \mathbf{x}}{\partial \boldsymbol{\xi}},$$

$$\begin{bmatrix} \frac{\partial f}{\partial \xi} & \frac{\partial f}{\partial \eta} & \frac{\partial f}{\partial \zeta} \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} & \frac{\partial f}{\partial z} \end{bmatrix} \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{bmatrix}. \tag{12}$$

The matrix $\mathbf{J} = \partial \mathbf{x}/\partial \boldsymbol{\xi}$ is known as the Jacobian of the mapping. Columns of $\mathbf{J}$ are tangent vectors to the logical coordinate system. The determinant of the Jacobian $J = |\mathbf{J}|$ relates volume elements between the logical and Cartesian coordinate systems

$$dx\,dy\,dz = J\,d\xi\,d\eta\,d\zeta. \tag{13}$$

The inverse of the Jacobian matrix is

$$\mathbf{J}^{-1} = \frac{\partial \boldsymbol{\xi}}{\partial \mathbf{x}} = \frac{1}{J} \begin{bmatrix} \begin{vmatrix} \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{vmatrix} & \begin{vmatrix} \frac{\partial z}{\partial \eta} & \frac{\partial x}{\partial \eta} \\ \frac{\partial z}{\partial \zeta} & \frac{\partial x}{\partial \zeta} \end{vmatrix} & \begin{vmatrix} \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} \end{vmatrix} \\ \begin{vmatrix} \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \end{vmatrix} & \begin{vmatrix} \frac{\partial z}{\partial \zeta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial x}{\partial \xi} \end{vmatrix} & \begin{vmatrix} \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \end{vmatrix} \\ \begin{vmatrix} \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \end{vmatrix} & \begin{vmatrix} \frac{\partial z}{\partial \xi} & \frac{\partial x}{\partial \xi} \\ \frac{\partial z}{\partial \eta} & \frac{\partial x}{\partial \eta} \end{vmatrix} & \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{vmatrix} \end{bmatrix} \tag{14}$$

Rows of $J\mathbf{J}^{-1}$ are surface area vectors $d\mathbf{S}$ for surfaces of constant ξ, η or ζ, and are formed by cross-products of the tangent vectors. Solving (12) gives the gradient of f:

$$\nabla f = \frac{\partial f}{\partial \boldsymbol{\xi}} \cdot \mathbf{J}^{-1}. \tag{15}$$

Because we seek the average gradient over the element volume, we will use the approximation

$$\nabla f \approx \frac{1}{V} \int_V \nabla f \, dV, \tag{16}$$

and define discrete operators D_x, D_y and D_z:

$$D_i F \equiv \int_{V^C} \frac{\partial f}{\partial x_i} \, dV, \tag{17}$$

$$D_i : H^N \to H^C, \tag{18}$$

where V^C is the volume of the central cell enclosed by the logical coordinates

$$\xi_m < \xi < \xi_{m+1},$$
$$\eta_m < \eta < \eta_{m+1},$$
$$\zeta_m < \zeta < \zeta_{m+1}. \tag{19}$$

Substituting (13), (15) and (19) into (17) gives

$$\hat{\mathbf{x}}D_x F + \hat{\mathbf{y}}D_y F + \hat{\mathbf{z}}D_z F = \int_\zeta \int_\eta \int_\xi \frac{\partial f}{\partial \boldsymbol{\xi}} \cdot \mathbf{J}^{-1} J \, d\xi \, d\eta \, d\zeta. \tag{20}$$

A computer algebra system is helpful for solving this integral. The remainder of this paper is concerned with the 3-D, linear case

($n = 1$). The resulting expressions for D_i are rather complex and are not available elsewhere, so we tabulate them in the Appendix. If elements are restricted in shape to rectangular parallelepipeds, the operators simplify to

$$(D_x F)_{000} = \frac{1}{4}(Z_1 - Z_0)(Y_1 - Y_0)$$
$$(F_{111} + F_{100} - F_{010} - F_{001}$$
$$-F_{000} - F_{011} + F_{101} + F_{110}), \tag{21}$$

$$(D_y F)_{000} = \frac{1}{4}(X_1 - X_0)(Z_1 - Z_0)$$
$$(F_{111} - F_{100} + F_{010} - F_{001}$$
$$-F_{000} + F_{011} - F_{101} + F_{110}) \tag{22}$$

and

$$(D_z F)_{000} = \frac{1}{4}(Y_1 - Y_0)(X_1 - X_0)$$
$$(F_{111} - F_{100} - F_{010} + F_{001}$$
$$-F_{000} + F_{011} + F_{101} - F_{110}), \tag{23}$$

each of which is recognizable as the average of four separate finite-difference operations along the cell edges.

Now that we have the difference operator D_i that operates on the nodal functions, the next task is to build a complementary difference operator that operates on a cell function and returns a nodal function:

$$\mathcal{D}_i : H^C \to H^N. \tag{24}$$

We will rely on the previously derived operators D_i and the adjoint relation between gradient and divergence. The goal is to ensure global conservation of the numerical scheme. This is the guiding principle of SOM and other mimetic methods. They attempt to 'mimic' fundamental conservation laws of the continuum. In this case D_i is called the natural operator and $\mathcal{D}_i$ is called the support operator.

Applying the divergence theorem to the product $f\mathbf{w}$ gives

$$\int_V (\nabla f) \cdot \mathbf{w} \, dV + \int_V f(\nabla \cdot \mathbf{w}) \, dV = \int_S f\mathbf{w} \cdot d\mathbf{S}. \tag{25}$$

When the normal component of $\mathbf{w}$ at the boundary surface is 0,

$$\int_V f\nabla \cdot \mathbf{w} \, dV = -\int_V (\nabla f) \cdot \mathbf{w} \, dV, \tag{26}$$

which expresses the fact that gradient and divergence are adjoint operators. The adjoint relationship has the discrete analogue

$$\sum_{i,j,k,l=0}^{3,p-1,q-1,r-1} (D_i F)_{jkl}(W_i)_{jkl} = -\sum_{i,j,k,l=0}^{3,p,q,r} F_{jkl}(\mathcal{D}_i W_i)_{jkl}, \tag{27}$$

where $\mathbf{W}_{jkl} \in H^C$ is a cell-valued vector function. Until this point we have considered only the local vicinity of the difference operator. Here we broaden the scope to the global problem domain, where the mesh size is $p \times q \times r$.

Inserting $D_i F$ into the left-hand side of (27) results in a summation factored in terms of common $\mathbf{W}_{jkl}$ values. We can refactor this summation in terms of common F_{jkl}. Equating this result to the right hand side of (27), we can write the expressions for $\mathcal{D}_i$ as listed in

the Appendix. On a rectangular mesh the equations simplify to

$$
\begin{aligned}
(\mathcal{D}_x \mathbf{W})_{111} = {} & \frac{1}{4}(Z_2 - Z_1)[(Y_2 - Y_1)(\mathbf{W}_{111} - \mathbf{W}_{011}) \\
& + (Y_1 - Y_0)(\mathbf{W}_{101} - \mathbf{W}_{001})] \\
& + \frac{1}{4}(Z_1 - Z_0)[(Y_2 - Y_1)(\mathbf{W}_{110} - \mathbf{W}_{010}) \\
& + (Y_1 - Y_0)(\mathbf{W}_{100} - \mathbf{W}_{000})],
\end{aligned}
\tag{28}
$$

$$
\begin{aligned}
(\mathcal{D}_y \mathbf{W})_{111} = {} & \frac{1}{4}(X_2 - X_1)[(Z_2 - Z_1)(\mathbf{W}_{111} - \mathbf{W}_{101}) \\
& + (Z_1 - Z_0)(\mathbf{W}_{110} - \mathbf{W}_{100})] \\
& + \frac{1}{4}(X_1 - X_0)[(Z_2 - Z_1)(\mathbf{W}_{011} - \mathbf{W}_{001}) \\
& + (Z_1 - Z_0)(\mathbf{W}_{010} - \mathbf{W}_{000})]
\end{aligned}
\tag{29}
$$

and

$$
\begin{aligned}
(\mathcal{D}_z \mathbf{W})_{111} = {} & \frac{1}{4}(Y_2 - Y_1)[(X_2 - X_1)(\mathbf{W}_{111} - \mathbf{W}_{110}) \\
& + (X_1 - X_0)(\mathbf{W}_{011} - \mathbf{W}_{010})] \\
& + \frac{1}{4}(Y_1 - Y_0)[(X_2 - X_1)(\mathbf{W}_{101} - \mathbf{W}_{100}) \\
& + (X_1 - X_0)(\mathbf{W}_{001} - \mathbf{W}_{000})].
\end{aligned}
\tag{30}
$$

At the boundaries, certain terms in $\mathcal{D}_i$ drop out. For example at the boundary $j = 0$, (28), (29) and (30) reduce to

$$
\begin{aligned}
(\mathcal{D}_x \mathbf{W})_{011} = {} & \frac{1}{4}(Z_2 - Z_1)[(Y_2 - Y_1)\mathbf{W}_{011} \\
& + (Y_1 - Y_0)\mathbf{W}_{001}] \\
& + \frac{1}{4}(Z_1 - Z_0)[(Y_2 - Y_1)\mathbf{W}_{010} \\
& + (Y_1 - Y_0)\mathbf{W}_{000}],
\end{aligned}
\tag{31}
$$

$$
\begin{aligned}
(\mathcal{D}_y \mathbf{W})_{011} = {} & \frac{1}{4}(X_2 - X_1)[(Z_2 - Z_1)(\mathbf{W}_{011} - \mathbf{W}_{001}) \\
& + (Z_1 - Z_0)(\mathbf{W}_{010} - \mathbf{W}_{000})]
\end{aligned}
\tag{32}
$$

and

$$
\begin{aligned}
(\mathcal{D}_z \mathbf{W})_{011} = {} & \frac{1}{4}(X_2 - X_1)[(Y_2 - Y_1)(\mathbf{W}_{011} - \mathbf{W}_{010}) \\
& + (Y_1 - Y_0)(\mathbf{W}_{001} - \mathbf{W}_{000})].
\end{aligned}
\tag{33}
$$

Accounting for each face, edge and corner of the mesh, there exist 48 different boundary operators for $\mathcal{D}_i$. The application of the boundary operators is simplified in practice by using a ghost cell technique. We extend the mesh outside of the boundaries with 'ghost' cells, and in those cells $\mathbf{W}_{jkl}$ is always zero. Applying the interior operators (28), (29) and (30) to the expanded mesh reproduces the boundary operators appropriately.

We can now approximate the partial derivative of F with

$$
\frac{(D_i F)_{jkl}}{V_{jkl}^C},
\tag{34}
$$

and approximate the partial derivative of $\mathbf{W}$ with

$$
\frac{(\mathcal{D}_i \mathbf{W})_{jkl}}{V_{jkl}^N},
\tag{35}
$$

where V_{jkl}^C is the cell volume and V_{jkl}^N is the nodal volume. The cell volume can be found by noting that

$$
\int_V \frac{\partial x_i}{\partial x_i} dV = V,
\tag{36}
$$

and differencing any one of the nodal coordinates

$$
V_{jkl}^C = (D_x X)_{jkl} = (D_y Y)_{jkl} = (D_z Z)_{jkl}.
\tag{37}
$$

The nodal volume is then found by averaging the surrounding eight cell volumes

$$
\begin{aligned}
V_{000}^N = {} & \frac{1}{8}\big(V_{000}^C + V_{011}^C + V_{101}^C + V_{110}^C \\
& + V_{111}^C + V_{100}^C + V_{010}^C + V_{001}^C\big).
\end{aligned}
\tag{38}
$$

3 ONE-POINT QUADRATURE

The gradient operator D_i is defined in eq. (17) by exactly integrating the integral (16). An alternative method is to approximate the integral with numerical quadrature. For one-point quadrature we simply evaluate eq. (15) at the centre of the stencil

$$
D_i F \equiv \left. \frac{\partial f}{\partial x_i} \right|_{\xi=(\xi_m + \xi_{m+1})/2}.
\tag{39}
$$

The resulting operators, eqs (A7)–(A12) listed in the Appendix, offer some computational savings. One-point quadrature has been frequently applied to seismic problems (see e.g. Day *et al.* 2005; Ma & Liu 2006). The operator D_i is often called the 'B matrix' in FEM literature. Goudreau & Hallquist (1982) found that for their applications, exactly integrated elements perform no better than one-point quadrature, and therefore, the extra computational expense is not justified. However, this may not be true for all applications. With non-parallelepiped elements, one-point quadrature fails the patch test. One consequence of this is that rigid body motion can lead to non-zero stress. To illustrate, from eqs (A7)–(A11), the one-point quadrature gradient $D_i F$ clearly evaluates to zero when F is uniform. However, the divergence $\mathcal{D}_i \mathbf{W}$, listed in eqs (A8)–(A12), does not in general evaluate to zero when $\mathbf{W}$ is uniform. The non-physical consequence is that energy can enter the system due simply to the shape of the mesh. Conversely, the exactly integrated operators, eqs (A1) through (A6) always evaluate to zero for uniform fields F and $\mathbf{W}$. We have not established whether, in practice, this is an important issue for seismic wave simulations, but violation of conservation of energy is clearly undesirable and should be avoided if possible.

The exactly integrated elements require roughly twice as many floating point operations to compute. However, the difference can be negated, and additional computational efficiency achieved, by holding the operators in memory rather than calculating them on-the-fly. One can store D_i in a memory array, and $\mathcal{D}_i$ can be had directly from D_i via the adjoint relation. Storage of D_i requires 24 memory variables per element, while storage of $\mathbf{X}$ for on-the-fly operators requires three variables per element. The 21 variable increase roughly doubles the amount of storage needed overall, while the number of floating point operations is reduced by a factor of 6 for the complete algorithm.

Tests of our code (summarized in Table 1), under ideal conditions with no parallel processing or file output, show a 25 per cent

Table 1. Resource usage per node per time time step for different operators. Numbers are for the complete wave simulation algorithm.

Operator type	Floating pt. ops.	Memory variables	Normalized run-time
Exactly integrated	3380	23	3.56
One-point quadrature	1688	23	2.67
Stored operators	518	44	1.27
Rectangular	446	20	1.00

reduction in runtime for one-point quadrature, and a factor of 3 reduction for pre-computed operators. This is only half of the speedup expected considering only floating point operations. However, memory access speed can be an equally important factor on architectures with fast CPUs. Parallel processing overhead and file access reduce even further the relative speed gains. Due to the modest performance penalty, and the potential for removing that penalty altogether by pre-computed operators, we prefer the exactly integrated elements.

4 NUMERICAL ALGORITHM

We can now build an explicit time stepping scheme by discretizing eqs (1)–(5). The continuous field variables are replaced with discrete variables of the same name. On the nodes we have $(\mathbf{u}, \mathbf{v}, \mathbf{a}, \rho, \gamma, \mathbf{g}) \in H^N$, and on the cells we have $(\sigma, \lambda, \mu) \in H^C$. Time is discretized with constant spacing Δt. Spatial derivative are approximated with the operators (34) and (35) and time derivatives are approximated with with second-order centred differences. The time index is indicated by a superscript, and for clarity, spatial indices are omitted in the discrete equations

$$g_{ij} = D_j\big(u_i^n + \gamma v_i^{n-1/2}\big), \tag{40}$$

$$\sigma_{ij} = \Lambda\, \delta_{ij} g_{kk} + M(g_{ij} + g_{ji}), \tag{41}$$

$$a_i = R\, \mathcal{D}_j\, \sigma_{ij}, \tag{42}$$

$$v_i^{n+1/2} = v_i^{n-1/2} + \Delta t\, a_i, \tag{43}$$

$$u_i^{n+1} = u_i^n + \Delta t\, v_i^{n+1/2}. \tag{44}$$

The material variables incorporate the cell and node volumes

$$R = \frac{1}{\rho V^N}, \tag{45}$$

$$\Lambda = \frac{\lambda}{V^C}, \tag{46}$$

$$M = \frac{\mu}{V^C}. \tag{47}$$

If material contrasts are to be aligned with the cell boundaries, it is most convenient to begin with an initial cell valued density function $\rho^C \in H^C$. Care must be taken to conserve global mass, and correctly align the material boundary when finding the nodal densities. This is achieved by weighting cell density by cell volume prior to averaging,

$$R_{000} = 8\big(V_{000}^C \rho_{000}^C + V_{011}^C \rho_{011}^C + V_{101}^C \rho_{101}^C + V_{110}^C \rho_{110}^C$$
$$+ V_{111}^C \rho_{111}^C + V_{100}^C \rho_{100}^C + V_{010}^C \rho_{010}^C + V_{001}^C \rho_{001}^C\big)^{-1}. \tag{48}$$

By the same principle, when material contrasts are to bisect the cells, it is most convenient to begin with initial nodal elastic moduli $(\lambda^N, \mu^N) \in H^N$, and compute the cell values by weighted harmonic averaging,

$$\Lambda_{111} = 8\left(\frac{V_{000}^N}{\lambda_{000}^N} + \frac{V_{011}^N}{\lambda_{011}^N} + \frac{V_{101}^N}{\lambda_{101}^N} + \frac{V_{110}^N}{\lambda_{110}^N}\right.$$
$$\left. + \frac{V_{111}^N}{\lambda_{111}^N} + \frac{V_{100}^N}{\lambda_{100}^N} + \frac{V_{010}^N}{\lambda_{010}^N} + \frac{V_{001}^N}{\lambda_{001}^N}\right)^{-1}, \tag{49}$$

$$M_{111} = 8\left(\frac{V_{000}^N}{\mu_{000}^N} + \frac{V_{011}^N}{\mu_{011}^N} + \frac{V_{101}^N}{\mu_{101}^N} + \frac{V_{110}^N}{\mu_{110}^N}\right.$$
$$\left. + \frac{V_{111}^N}{\mu_{111}^N} + \frac{V_{100}^N}{\mu_{100}^N} + \frac{V_{010}^N}{\mu_{010}^N} + \frac{V_{001}^N}{\mu_{001}^N}\right)^{-1}. \tag{50}$$

Stability of the explicit scheme requires that Δt be less than the shortest time for waves to traverse a cell. The condition

$$1 < \frac{\Delta t}{\Delta x} \sqrt{\frac{3(\lambda + 2\mu)}{\rho}} \tag{51}$$

is necessary to ensure stability for a rectangular mesh of constant spacing Δx, and can be used as an approximate guide for selecting stable time steps for non-rectangular meshes as well.

The 24 degrees of freedom for displacement on the eight node hexahedral element can be decomposed into three rigid body, nine uniform, and 12 non-uniform strain modes. The non-uniform modes are alternatively referred to as hourglass, keystone or bending modes. Hourglass modes are orthogonal to and transparent to single-point derivative operators such as we have derived. Unchecked they may grow to dominate the solution, so they must be independently controlled. We use a modified form of the hourglass control scheme described by Flanagan & Belytschko (1981) and more recently by Day *et al.* (2005) and Ma & Liu (2006). We define the hourglass operators

$$Q_i : H^N \to H^C, \tag{52}$$

$$(Q_1 F)_{000} = F_{000} + F_{011} - F_{101} - F_{110}$$
$$+ F_{111} + F_{100} - F_{010} - F_{001}, \tag{53}$$

$$(Q_2 F)_{000} = F_{000} + F_{101} - F_{110} - F_{011}$$
$$+ F_{111} + F_{010} - F_{001} - F_{100}, \tag{54}$$

$$(Q_3 F)_{000} = F_{000} + F_{110} - F_{011} - F_{101}$$
$$+ F_{111} + F_{001} - F_{100} - F_{010}, \tag{55}$$

$$(Q_4 F)_{000} = F_{000} + F_{011} + F_{101} + F_{110}$$
$$- F_{111} - F_{100} - F_{010} - F_{001}, \tag{56}$$

and

$$\mathcal{Q}_i : H^C \to H^N, \tag{57}$$

$$(\mathcal{Q}_1 \mathbf{W})_{111} = \mathbf{W}_{111} + \mathbf{W}_{100} - \mathbf{W}_{010} - \mathbf{W}_{001}$$
$$+ \mathbf{W}_{000} + \mathbf{W}_{011} - \mathbf{W}_{101} - \mathbf{W}_{110}, \tag{58}$$

$$(\mathcal{Q}_2 \mathbf{W})_{111} = \mathbf{W}_{111} + \mathbf{W}_{010} - \mathbf{W}_{001} - \mathbf{W}_{100}$$
$$+ \mathbf{W}_{000} + \mathbf{W}_{101} - \mathbf{W}_{110} - \mathbf{W}_{011}, \tag{59}$$

$$(\mathcal{Q}_3 \mathbf{W})_{111} = \mathbf{W}_{111} + \mathbf{W}_{001} - \mathbf{W}_{100} - \mathbf{W}_{010}$$
$$+ \mathbf{W}_{000} + \mathbf{W}_{110} - \mathbf{W}_{011} - \mathbf{W}_{101}, \tag{60}$$

$$(\mathcal{Q}_4 \mathbf{W})_{111} = \mathbf{W}_{111} + \mathbf{W}_{100} + \mathbf{W}_{010} + \mathbf{W}_{001}$$
$$- \mathbf{W}_{000} - \mathbf{W}_{011} - \mathbf{W}_{101} - \mathbf{W}_{110}. \tag{61}$$

Viscous as well as stiffness hourglass control may be used, for which we define the viscosity β, and stiffness

$$\mathcal{Y} = \frac{\mu(\lambda + \mu)}{6(\lambda + 2\mu)}. \tag{62}$$

The correction is applied by modifying the acceleration eq. (42), and the discrete equations now become:

$$g_{ij} = D_j\big(u_i^n + \gamma v_i^{n-1/2}\big), \tag{63}$$

$$\sigma_{ij} = \Lambda \, \delta_{ij} g_{kk} + M(g_{ij} + g_{ji}), \tag{64}$$

$$a_i = R \, \mathcal{D}_j \, \sigma_{ij} - \mathcal{Q}_k \, \mathcal{Y} \, Q_k\big(u_i^n + \beta v_i^{n-1/2}\big), \tag{65}$$

$$v_i^{n+1/2} = v_i^{n-1/2} + \Delta t \, a_i, \tag{66}$$

$$u_i^{n+1} = u_i^n + \Delta t \, v_i^{n+1/2}. \tag{67}$$

The form we choose for hourglass stiffness $\mathcal{Y}$ is based on the approximate analysis of Kosloff & Frazier (1978). They found that the growth of the hourglass modes in 2-D grids was effectively resisted with a stiffness of this form, and numerical experiments (e.g. Day *et al.* 2005; Ma & Liu 2006) demonstrate that the same stiffness works well to suppress growth of the corresponding 3-D modes, especially in combination with a damping β of order 1 (and results are not very sensitive to values of $\mathcal{Y}$ and β, once they are large enough to suppress mode growth).

5 PERFECTLY MATCHED LAYER

Modelling waves in a boundless material requires artificial truncation of the computational domain. Various types of absorbing boundaries have been used to suppress artificial reflections at the boundaries. One of the most effective is the method of PML. First introduced for electromagnetic waves by Berenger (1994, 1996), PML sets up an absorbing layer where waves are exponentially attenuated and the reflection coefficient at the layer interface is nearly zero for all angles of incidence. Marcinkovich & Olsen (2003) present a PML absorbing boundary condition for the velocity–stress formulation of elastodynamics. We modify their given system of equations to find an equivalent formulation that offers a compact notation and that is more optimized numerically. The modification is a reordering of operations that results in damping of the spatial derivatives of velocity and stress rather velocity and stress themselves. The change reduces the additional required storage from nine to six variables per damping direction, and reduces the number of multiplication operations. The modified formulation is

$$\dot{g}_{ij} + d(x_j)g_{ij} = \partial_j v_i, \tag{68}$$

$$\dot{\sigma}_{ij} = \lambda \, \delta_{ij} \sum_k \dot{g}_{kk} + \mu(\dot{g}_{ij} + \dot{g}_{ji}), \tag{69}$$

$$\dot{p}_{ij} + d(x_j)p_{ij} = \partial_j \sigma_{ij}, \tag{70}$$

$$\dot{v}_i = \frac{1}{\rho} \sum_j \dot{p}_{ij}, \tag{71}$$

where $d(x_j)$ is the damping profile, and x_j is the distance measured from the node or cell location to the PML interface along the x, y or z direction. Note that repeated indices do not imply summation here. In this formulation, the PML interface plane must be normal to one of the Cartesian directions. Where PML zones overlap, such as at the

corners of the model, damping occurs in more than one direction. For the interior of the model, not in the PML, $d(x_i) = 0$, and the equations reduce to the elastic wave equations. This formulation is suitable for numerical schemes that store the elastic state as velocity and *stress*. In this paper, however, we have developed a scheme that stores velocity and *displacement*, so the system must be modified slightly to

$$\dot{g}_{ij} + d(x_j)g_{ij} = \partial_j v_i, \tag{72}$$

$$\sigma_{ij} = \lambda \, \delta_{ij} \sum_k g_{kk} + \mu(g_{ij} + g_{ji}), \tag{73}$$

$$\dot{p}_{ij} + d(x_j)p_{ij} = \partial_j \sigma_{ij}, \tag{74}$$

$$a_i = \frac{1}{\rho} \sum_j \dot{p}_{ij}, \tag{75}$$

$$\dot{v}_i = a_i, \tag{76}$$

$$\dot{u}_i = v_i. \tag{77}$$

Discretizing eqs (72) and (74) gives

$$\frac{g_{ij}^n - g_{ij}^{n-1}}{\Delta t} + d(x_i)\frac{g_{ij}^n + g_{ij}^{n-1}}{2} = D_j v_i^{n-1/2} \tag{78}$$

and

$$\frac{p_{ij}^{n+1/2} - p_{ij}^{n-1/2}}{\Delta t} + d(x_i)\frac{p_{ij}^{n+1/2} + p_{ij}^{n-1/2}}{2} = \mathcal{D}_j \sigma_{ij}, \tag{79}$$

from which we can build an explicit time stepping scheme for the PML similar to that for the viscoelastic solid in eqs (63)–(67):

$$g_{ij}^n = \frac{2\Delta t}{2 + d(x_j)\Delta t} \, D_j v_i^{n-1/2} + \frac{2 - d(x_j)\Delta t}{2 + d(x_j)\Delta t} \, g_{ij}^{n-1}, \tag{80}$$

$$\sigma_{ij} = \Lambda\delta_{ij} \sum_k g_{kk}^n + M\big(g_{ij}^n + g_{ji}^n\big), \tag{81}$$

$$\dot{p}_{ij} = \frac{2}{2 + d(x_i)\Delta t}\mathcal{D}_j \sigma_{ij} - \frac{2d(x_i)}{2 + d(x_i)\Delta t} p_{ij}^{n-1/2}, \tag{82}$$

$$a_i = R \sum_j \dot{p}_{ij} - \mathcal{Q}_k \, \mathcal{Y} \, Q_k\big(\beta v_i^{n-1/2}\big), \tag{83}$$

$$p_{ij}^{n+1/2} = p_{ij}^{n-1/2} + \Delta t \, \dot{p}_{ij}, \tag{84}$$

$$v_i^{n+1/2} = v_i^{n-1/2} + \Delta t \, a_i, \tag{85}$$

$$u_i^{n+1} = u_i^n + \Delta t \, v_i^{n+1/2}. \tag{86}$$

For each direction of damping, six extra memory arrays are required for the storage of g_{1j}, g_{2j}, g_{3j}, p_{1j}, p_{2j} and p_{3j}. The non-damped components of g_{ij} and p_{ij} need not be stored because, when $d(x_j) = 0$, eq. (80) can be replaced by

$$g_{ij} = D_j u_i^n, \tag{87}$$

and eq. (82) reduces to

$$\dot{p}_{ij} = \mathcal{D}_j \sigma_{ij}^n. \tag{88}$$

Ma & Liu (2006) have found that stiffness hourglass control can cause instability in the PML. Therefore, only viscous hourglass control is used in eq. (83).

Following Marcinkovich & Olsen, the damping profile within the PML zone is given by

$$d(x) = \frac{3.5V_s}{w}\left(\frac{x}{w}\right)^2\left(\frac{8}{15}n - \frac{3}{100}n^2 + \frac{1}{1500}n^3\right), \tag{89}$$

where w is the PML thickness, n is the number of mesh nodes in the PML, and V_s is the harmonic mean of the minimum and maximum S-wave velocities present in the material model.

6 LAYERED MODEL TEST

For numerical verification of the SOM, we reproduce the double-couple point source test LOH.1 from Day & Bradley (2001). The model, diagrammed in Fig. 3, consists of a 1 km layer over a uniform half-space. In the layer $V_s = 2000 \text{ m s}^{-1}$, $V_p = 4000 \text{ m s}^{-1}$ and density $\rho = 2600 \text{ kg m}^{-3}$, and in the underlying half-space $V_s = 3464 \text{ m s}^{-1}$, $V_p = 6000 \text{ m s}^{-1}$ and $\rho = 2700 \text{ kg m}^{-3}$. The model is purely elastic, so viscosity $\gamma = 0$, but we use a relatively high hourglass viscosity $\beta = \Delta t$. The time step length $\Delta t = 0.004$ s.

We do two calculations, one with a rectangular mesh of node spacing $\Delta x = 50$ m, and another with a mesh highly distorted by shearing. The sheared mesh is constructed from the rectangular mesh by applying the coordinate mapping

$$\begin{bmatrix} X'_{jkl} \\ Y'_{jkl} \\ Z'_{jkl} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{jkl} \\ Y_{jkl} \\ Z_{jkl} \end{bmatrix}, \tag{90}$$

where (X, Y, Z) are coordinates of the rectangular mesh and (X', Y', Z') are coordinates of the sheared mesh. The mapping is a superposition of two 45° simple shears with a maximum shear angle of $\tan^{-1}(\sqrt{2}) = 54.7°$. This mesh is by no means a test for all types of possible mesh distortion. It does not, for example, address the case of non-parallelepiped elements or element volume variability. However, sheared meshes are useful for a number of types of earthquake problems, and are readily compatible with the layered model. A PML absorbing boundary is used for the bottom and side surfaces of the rectangular mesh and the bottom surface of the sheared mesh. Since our PML implementation is limited to boundary surfaces that are normal to the Cartesian directions, it cannot be used on the sides

of the sheared mesh. One way around this is to gradually reduce the shear of the mesh to zero at the boundary. We have taken an alternative approach by simply extending the mesh far enough that artificial reflections to not return during the simulation time.

A double-couple point source is located at the coordinates $(0, 0, 2 \text{ km})$. The non-zero components of the moment tensor are

$$M_{xy} = M_{yx} = \frac{M_0 H(t)}{\sqrt{2\pi}S}e^{-0.5(t-4S)^2/S^2}, \tag{91}$$

where $M_0 = 10^{18}$ N m, $S = 0.09$ s and $H(t)$ is a step function. The half-width of the Gaussian source spectrum is 2.08 Hz, corresponding to a Rayleigh wavelength of 17 gridpoints, and the source spectrum falls to 10 per cent of its maximum at 3.8 Hz, or about 9 points per Rayleigh wavelength. The source is inserted into the wave simulation by normalizing the moment tensor by cell volume and adding it to the stress tensor after eq. (64).

Particle velocity is recorded at three receiver locations at the surface: S1 $(-6 \text{ km}, -8 \text{ km}, 0)$, S2 $(6 \text{ km}, -8 \text{ km}, 0)$ and S3 $(6 \text{ km}, 8 \text{ km}, 0)$. At 2.08 Hz, the source–receiver distance is approximately equal to 12 Rayleigh wavelengths in the layer, and 3.5 P wavelengths in the half-space, and at 3.8 Hz, the distance is 21 Rayleigh wavelengths, and 6.3 P wavelengths. Due to symmetry, the analytical solutions for S1 and S3 are the same in cylindrical coordinates (with z axis through the source point). Receiver S2 is also the same with a sign change. For the numerical solution, though, the path from the source to each receiver has a different axis of propagation through the geometry of the sheared elements. The ray path to S1 is aligned with the long axis of the elements, while the path to S2 is aligned with the short axis. Receiver S3 has an intermediate alignment. In this way, anisotropy of wave propagation due to element distortion can be tested.

Fig. 4 compares velocity time histories from both the rectangular- and sheared-mesh calculations against a reference solution computed by the frequency-wavenumber (FK) method of Apsel & Luco (1983). The reference solution can be viewed as semi-analytical, in that the FK method first forms the doubly Fourier transformed (from time and radial coordinate to frequency and wavenumber) solution analytically, then inverts the transformed solution to space–time coordinates by numerical quadrature. The radial and vertical components show good agreement for early arriving P waves and late
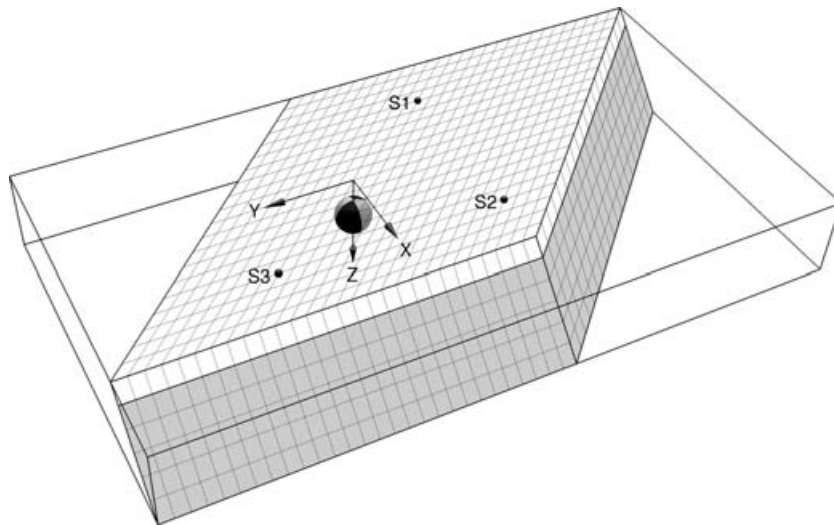

Figure 3. Perspective view of the layer over half-space model on a sheared mesh. The layer is 1 km thick. The source is located at 2 km depth, beneath the origin. Observation points are marked S1, S2 and S3.
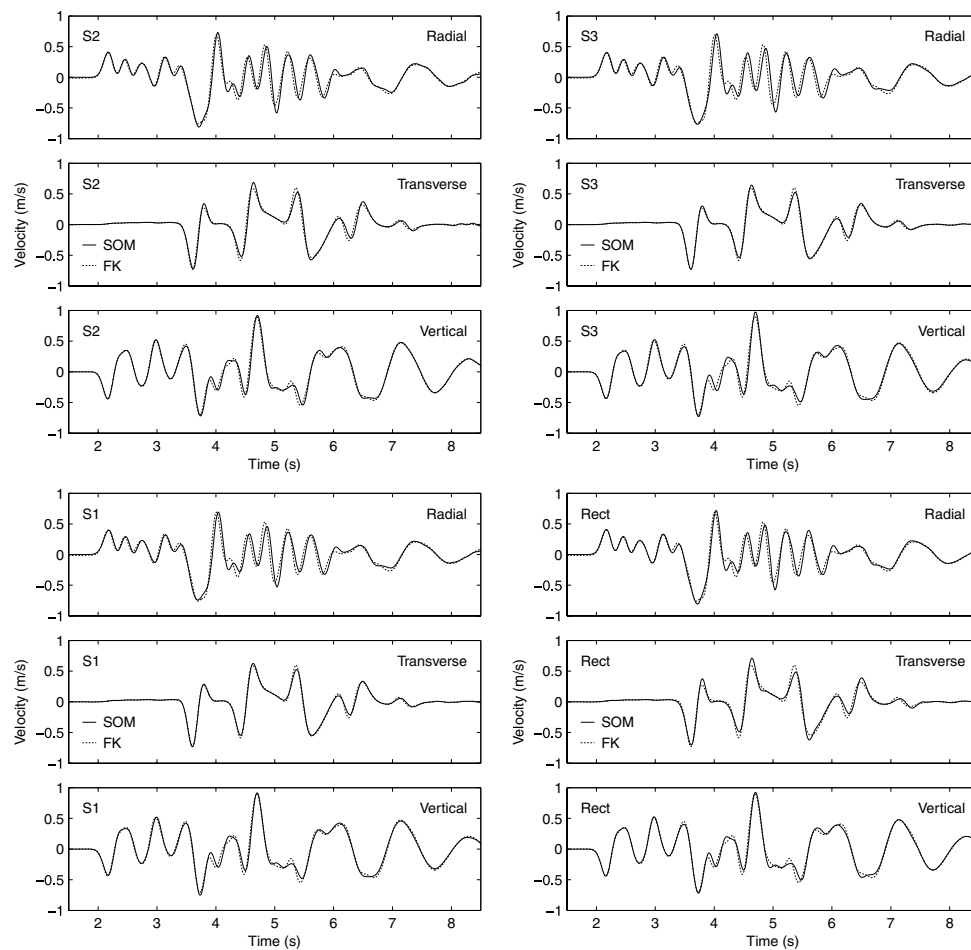
Figure 4. Simulated ground velocity compared to the reference solution calculated by frequency-wavenumber integration (FK). For comparison purposes, the polarity of S2 has been inverted to match stations S1 and S3.

arriving Rayleigh waves, with amplitude misfits up to 1 per cent of peak velocity, and arrival time misfits up to 0.01 s. Multiply reflected *P* waves have greater misfits of up to 10 per cent of peak velocity, and up to 0.04 s in time. All of the SOM curves have roughly the same level of misfit to the FK solution, and a high level of agreement with each other. This indicates that mesh shearing is not the main source of error for *P* waves and Rayleigh waves. Given the source bandwidth noted above, the level of waveform agreement achieved is consistent with expectations for a second-order accurate method, for which points-per-wavelength requirements in the range 10–15 are typically cited. The numerical errors take the form of non-physical dispersion, leading to phase velocity errors that (for second-order methods) increase monotonically with frequency (e.g. Virieux 1986) so of course the actual points-per-wavelength requirement depends upon how much tolerance for phase errors one has in a given application.

The effects of mesh anisotropy are most apparent for *S*-wave arrivals in the transverse component. Interestingly, the sheared mesh curves consistently match FK better than does the rectangular mesh curve. For each arrival, S1 is the closest match, followed by S3, then S2, and finally the rectangular case. We have observed that the differences between the stations increases with hourglass viscosity β, and we attribute the higher mismatch for the rectangular mesh to hourglass oscillations, and imperfect removal of them by the hourglass correction scheme. Rectangular meshes are more susceptible

to hourglass errors due to stronger coherence of the grid oscillation modes. Conversely, a greater diversity of nodal spacing and relative position lead to reduced hourglass mode coherence in the sheared mesh.

7 SEMI-CIRCULAR CANYON TEST

Our second numerical test aims to verify the free surface boundary condition in the presence of topographic features. A vertically incident *P* wave on a semi-circular canyon presents a challenging test for numerical methods, as significant energy is converted to SV and Rayleigh waves, and relative amplification is highly variable in and around the canyon. The problem has been thoroughly studied using various types of boundary integral methods that we may use to verify the results of our present method.

The problem is set up with dimensionless parameters $\rho = 1$, $V_p = 2$, $V_s = 1$, $\gamma = 0$ and a semi-circular cavity of radius $R_0 = 1$ at the surface of a half space (Fig. 5). We accommodate the 2-D geometry by constructing a mesh with a thickness of one element in the *z* direction, and restricting motion to the *x*–*y* plane. To facilitate planar boundary conditions, the outer boundaries are made rectangular. The dimensions of the mesh are $22R_0 \times 11R_0$. Elements gradually increase in size from $0.005R_0$ at the canyon to $0.07R_0$ at the boundaries. Computational savings are realized by calculating solutions
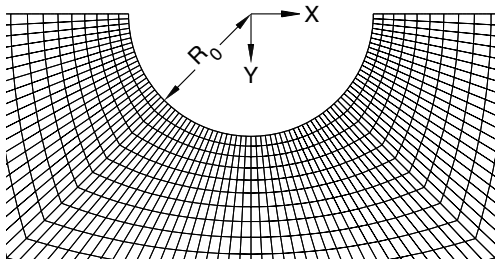
Figure 5. Close-in view of the mesh for a semi-circular canyon. The mesh extends outward to rectangular boundaries at $x = \pm 11 R_0$ and $y = 11 R_0$.

for only the positive x half of the mesh, and enforcing a symmetry boundary condition along the y-axis. The symmetry boundary condition is also applied to the $x = 11 R_0$ boundary, which is placed far enough from the canyon, that no horizontal motion is received during the simulation time. The vertically incident P wave is introduced as a boundary condition on displacement at the $y = 11 R_0$ boundary. We mention that the time function of the source is a Ricker wavelet of $f_0 = 0.5$, though the exact form is not crucial for the method of analysis using spectral ratios. The computation is run for 6000 iterations with a time step of $\Delta t = 0.002$.

Analysis is preformed in terms of normalized frequency $f_0 = \omega R_0 / 2\pi V_s$. We consider the displacement amplitude at the surface for two frequencies: $f_0 = 0.25$ and 1.0. The low frequency case, studied by Wong (1982), Sánchez-Sesma *et al.* (1985) and Mossessian & Dravinski (1987), corresponds to a P wavelength of four times the canyon width and an S wavelength of twice the canyon width. The high frequency case, also studied by Wong, as well as Kawase (1988) and Sánchez-Sesma & Campillo (1991), corresponds to a P wavelength equal to the canyon width, and an S wavelength equal to half of the canyon width. Our results agree particularly well with the more recent studies (Fig. 6).

For the high frequency case, the limiting factor for satisfactory agreement, is the spatial resolution of the P wave at the bottom boundary where elements are largest. For $f_0 = 1.0$, the resolution is 28 points per wavelength at the bottom. We have seen notable differences in the surface response, when the resolution goes below 20 points per wavelength for the P wave source. The $f = 0.25$ signal is more than adequately resolved in the discretization, with at least 50 points per wavelength for all types of waves.

8 PARALLELIZATION

For large computational tasks, code parallelization is required for maximal utilization of modern computer hardware. A significant benefit of the structured mesh approach we have taken is that apportioning work among multiple processors is straightforward. We have implemented domain decomposition, where the mesh is subdivided and distributed across processors. During each time step, between eqs (65) and (66) in the computational cycle, the nodal acceleration field at each subdomain edges is transmitted to the neighbouring subdomain. In this way, the subdomains are connected to form a single global computational domain.

To test parallel performance, we benchmarked our code on the TeraGrid IA64 cluster at the San Diego Supercomputer Center. A traditional strong scaling test measures runtime speedup for a fixed size problem as the number of processors increases. However, due to the distributed memory architecture of the cluster, it is impossible for a fixed size problem to fit in a single processor's memory and
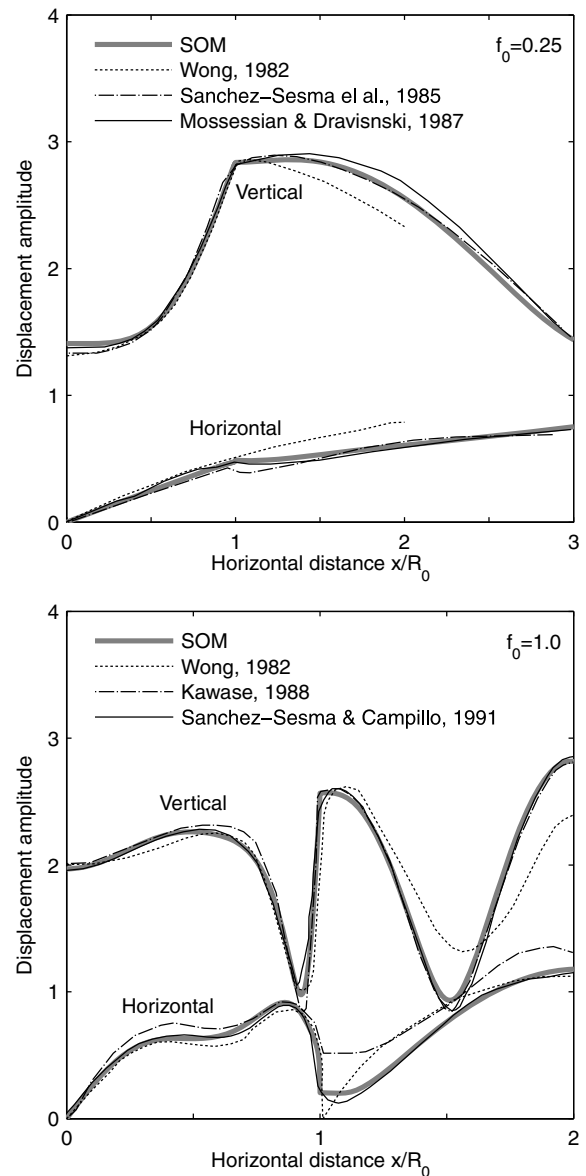

Figure 6. Surface displacement amplitude, at frequencies $f_0 = 0.25$ (top) and $f_0 = 1.0$ (bottom), as function of horizontal distance from the centre of the canyon. Amplitudes are relative to the vertically incident *P*-wave source. Results digitized from previous studies are shown for comparison.

at the same time be a representative test for the communication and memory access patterns of a typical real-world, large-scale problem. For this reason, we prefer a weak scaling test in which the problem size is increased proportionally to the number of processors. Our weak scaling benchmark consists of a problem size of $128 \times 128 \times 128$ nodes per processor, run for 10 time steps. Timing begins after code initialization, since, though insignificant for a long run, initialization time might well influence a short-running benchmark test. The tests take about 30 s to run on TeraGrid. Ideal weak scaling occurs when runtime stays the same as the number of processors (and problem size) increases. The code achieves 84 per cent efficiency on 512 processors. And, as shown in Table 2, the falloff in efficiency between 8 and 512 processors is very gradual, indicating the code likely will continue to scale well to higher numbers of processors.

Table 2. Weak scaling benchmark for the SDSC TeraGrid IA64 Cluster.

Processors	Normalized run-time	Efficiency
1	1.000	100 per cent
8	1.176	85.1 per cent
64	1.182	84.6 per cent
512	1.188	84.2 per cent

9 DISCUSSION AND CONCLUSIONS

We have described a 3-D, arbitrary-order, SOM for viscoelastic wave simulation and implemented it for the second order case. We have also implemented hourglass corrections, a free surface boundary condition, and a PML absorbing boundary condition. Our method addresses the problem of adding topography and non-planar boundaries to earthquake simulations. It has been tested against FK solutions for wave propagation in a layer over half-space model (LOH.1), on a rectangular mesh as well as a mesh deformed by $54.7°$ simple shear. For wavelengths long enough to be propagated without significant numerical dispersion (e.g. represented by more than 10 points per wavelength) in the rectangular mesh, shearing the mesh introduces no measurable anisotropy. The points-per-wavelength requirement found here is typical of second-order methods. The use of a structured mesh permits very efficient parallel execution, and we have been able to demonstrate scalability of the algorithm at greater than 80 per cent efficiency on up to 512 processors.

The choice of using a logically rectangular mesh rather than an unstructured mesh has trade-offs. Unstructured meshes, used by some of the other methods cited in Section 1, are more adaptable to arbitrary problem geometry. They also allow for a high degree of local mesh refinement, which is particularly useful for modelling basins with a large range of material velocities, or for resolving small-scale detail of fault rupture dynamics. On the other hand, structured meshes, though significantly more restrictive, are more computationally efficient. If the differentiation operators are saved in memory, floating-point operations are reduced by a factor of 6, nearly to the level of rectangular finite differences. This reduces actual run times by a factor of 3. Structured meshes also greatly simplify the tasks of parallelization, mesh generation and visualization of results. When the ratio of maximum to minimum wave speed over the problem domain is not extreme, and the problem geometry can be adequately handled, structured meshes are quite advantageous. As an example, we have run wave propagation problems of 1.8 billion nodes, using 1920 processors and 250 GB of memory, that achieved a computation rate of 370 Gflop s^{-1}. That example used the SCEC Community Velocity Model (Magistrale *et al.* 2000), with a ratio of maximum to minimum wave speed of roughly 20.

Our method can be easily adapted to split nodes and non-linear boundary conditions to simulate rupture. We presented rupture dynamics simulations on sheared meshes in Ely *et al.* (2005) that achieved accuracy comparable to rectangular mesh rupture dynamics methods. This will facilitate research on the dynamics of faults with realistic morphology, which, among other things, may produce strong nonlinear coupling between shear and normal stress perturbations. It also facilitates research on dynamics of dipping faults.

Because the method works explicitly with stress components (rather than a stiffness matrix, for example), it is easily generalized to calculate stresses from inelastic constitutive models (e.g. Coulomb plasticity, damage rheology, etc.), at the cost of saving the stress ten-

sor components globally. This capability would be important in the study earthquake rupture dynamics, for example, as rupture-induced off-fault nonlinear deformation may be a significant contributor to the energy budget of earthquakes, and may also influence strong motion amplitudes.

We have formulated the SOM method for arbitrary order of accuracy, leaving the potential for developing a higher order algorithm. At higher than order two, however, it may not be possible to solve integral (20) algebraically. Instead, numerical quadrature would be required to calculate the operator weights. Either method of calculating the weights will be computationally costly, and the optimal scheme may be to store the weights rather that recalculate them on-the-fly. In a stored scheme, fourth-order operators, with a $4 \times 4 \times 4$ stencil, are eight times more costly than second-order operators in terms of storage and computations per time step required to compute the spatial derivatives. When the entire numerical algorithm is considered, this translates to a fivefold increase in storage and computations for going to fourth order. The increased accuracy, though, reduces the points-per-wavelength requirement, permitting spatial and temporal resolution to be reduced. If the points-per-wavelength requirement can be reduced enough to surpass the break-even point in the cost trade-off, then the fourth-order method will be beneficial. Given that the memory usage goes like the third power of spatial resolution, and computational cost goes like the fourth power, we would need a factor of $\sqrt[3]{5}$ points-per-wavelength gain to break even on storage, and a factor of $\sqrt[4]{5}$ gain to break even on computations.

REFERENCES

Aagaard, B.T., 1999. Finite-element simulations of earthquakes, Technical Report EERL-99-03, California Institute of Technology, Earthquake Engineering Research Laboratory, Pasadena, CA.

Apsel, R.J. & Luco, J.E., 1983. On the Green's functions for a layered half-space. Part II, *Bull. seism. Soc. Am.,* **73**(4), 931–951.

Berenger, J.-P., 1994. A perfectly matched layer for the absorption of electromagnetic waves, *J. Comput. Phys.,* **114**(2), 185–200.

Berenger, J.-P., 1996. Three-dimensional perfectly matched layer for the absorption of electromagnetic waves, *J. Comput. Phys.,* **127**(2), 363–379.

Cruz-Atienza, V.M., 2006. Rupture Dynamique des Failles Non-Planaires en Différences Finies, *PhD thesis*, Université de Nice Sophia Antipolis.

Dalguer, L.A. & Day, S.M., 2007. Staggered-grid split-node method for spontaneous rupture simulation, *J. geophys. Res.,* **112**, B02302.

Day, S.M. & Bradley, C.R., 2001. Memory-efficient simulation of anelastic wave propagation, *Bull. seism. Soc. Am.,* **91**(3), 520–531.

Day, S.M. & Minster, J.-B., 1984. Numerical simulation of attenuated wavefields using a padé approximant method, *Geophys. J. Int.,* **78**(1), 105–118.

Day, S.M., Dalguer, L.A., Lapusta, N. & Liu, Y., 2005. Comparison of finite difference and boundary integral solutions to three-dimensional spontaneous rupture, *J. geophys. Res.,* **110**, B12307.

Dumbser, M. & Kašer, 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.

Ely, G.P., Minster, J.-B. & Day, S.M., 2005. SORD: a new rupture dynamics modeling code, *EOS, Trans. Am. geophys. Un.,* **86**(52), Fall Meet. Suppl., Abstract S43A–1067.

Flanagan, D.P. & Belytschko, T., 1981. A uniform strain hexahedron and quadrilateral with orthogonal hourglass control (in one-point integration of isoparametric finite element analysis), *Int. J. Numer. Methods Eng.,* **17**(5), 679–706.

Graves, R.W., 1996. Simulating seismic wave propagation in 3D elastic media using staggered-grid finite differences, *Bull. seism. Soc. Am.,* **86**(4), 1091–1106.

Goudreau, G.L. & Hallquist, J.O., 1982. Recent developments in large-scale finite element lagrangian hydrocode technology, *Comput. Method. Appl. M.,* **33**(1-3), 725.

Kase, Y. & Day, S.M., 2006. Spontaneous rupture processes on a bending fault, *Geophys. Res. Lett.,* **33**, L10302.

Kawase, H., 1988. Time-domain response of a semi-circular canyon for incident SV, P, and Rayleigh waves calculated by the discrete wavenumber boundary element method, *Bull. seism. Soc. Am.,* **78**(4), 1415–1437.

Komatitsch, D. & Tromp, J., 1999. Introduction to the spectral element method for three-dimensional seismic wave propagation, *Geophys. J. Int.,* **139**(3), 806–822.

Kosloff, D. & Frazier, G.A., 1978. Treatment of hourglass patterns in low order finite element codes, *Int. J. Numer. Anal. Methods Geomech.,* **2**(1), 57–72.

Levander, A.R., 1988. Fourth-order finite-difference P-SV seismograms, *Geophysics,* **53**(11), 1425–1436.

Ma, S. & Liu, P., 2006. Modeling of the perfectly matched layer absorbing boundaries and intrinsic attenuation in explicit finite-element methods, *Bull. seism. Soc. Am.,* **96**(5), 1779–1794.

Ma, S., Archuleta, R.J. & Page, M.T., 2007. Effects of large-scale surface topography on ground motion: an example of the San Gabriel mountains, Los Angeles, California, *Bull. seism. Soc. Am.,* in press.

Magistrale, H., Day, S.M., Clayton, R.W. & Graves, R.W., 2000. The SCEC Southern California reference three-dimensional seismic velocity model version 2, *Bull. seism. Soc. Am.,* **90**(6B), S65–76.

Marcinkovich, C. & Olsen, K., 2003. On the implementation of perfectly matched layers in a three-dimensional fourth-order velocity-stress finite difference scheme, *J. geophys. Res.,* **108**(B5), 2276.

Moczo, P., Robertson, J. & Eisner, L., 2006. The finite-difference time-domain method for modeling of seismic wave propagation, in *Advances in Wave Propagation in Heterogeneous Earth*, Vol. 48, pp. 421–516, eds Wu R.-S., Maupin, V. & Dmowska, R., Advances in Geophysics, Elsevier, Amsterdam.

Mossessian, T.K. & Dravinski, M., 1987. Application of a hybrid method for scattering of P, SV, and Rayleigh waves by near-surface irregularities, *Bull. seism. Soc. Am.,* **77**(5), 1784–1803.

Oglesby, D.D., 1999. Earthquake Dynamics on Dip-Slip Faults, *PhD thesis*, University of California, Santa Barbara.

Oglesby, D.D., Archuleta, R.J. & Nielsen, S.B., 2000. The three-dimensional dynamics of dipping faults, *Bull. seism. Soc. Am.,* **90**(3), 616–628.

Olsen, K.B. *et al.*, 2006. Strong shaking in Los Angeles expected from southern San Andreas earthquake, *Geophys. Res. Lett.,* **33**(L07305).

Rojas, O., Day, S.M., Castillo, J.E. & Dalguer, L.A., 2007. Modeling of rupture propagation using high-order mimetic finite differences, *Geophys. J. Int.,* in press.

Samarskii, A., Tishkin, V., Favorskii, A. & Shashkov, M., 1981. Operational finite-difference schemes, *Diff. Eqns.,* **17,** 854–862.

Samarskii, A., Tishkin, V., Favorskii, A. & Shashkov, M., 1982. Employment of the reference-operator method in the construction of finite difference analogs of tensor operations, *Diff. Eqns.,* **18,** 881–885.

Sánchez-Sesma, F.J. & Campillo, M., 1991. Diffraction of P, SV, and rayleigh waves by topographic features: a boundary integral formulation, *Bull. seism. Soc. Am.,* **81**(6), 2234–2253.

Sánchez-Sesma, F.J., Bravo, M.A. & Herrera, I., 1985. Surface motion of topographical irregularities for incident P, SV, and Rayleigh waves, *Bull. seism. Soc. Am.,* **75**(1), 263–269.

Shashkov, M., 1996. *Conservative Finite-Difference Methods on General Grids*, CRC Press, Boca Raton, FL.

Virieux, J., 1986. P-SV wave propagation in heterogeneous media: velocity-stress finite-difference method, *Geophysics,* **51**(4), 889–901.

Wong, H.L., 1982. Effect of surface topography on the diffraction of P, SV, and Rayleigh waves, *Bull. seism. Soc. Am.,* **72**(4), 1167–1183.

APPENDIX A: SPATIAL DIFFERENCE OPERATORS

These are the 3-D, second-order difference operators for gradient and divergence. Eqs (A1)–(A6) are the exactly integrated versions, and eqs (A8)–(A12) are the one-point quadrature versions.

Exactly integrated gradient, *x* component:

$$(D_x F)_{000} = \frac{1}{12}$$

$$\{F_{111}[(Y_{100} - Y_{011})(Z_{110} - Z_{101}) + Y_{011}(Z_{001} - Z_{010})$$
$$+ (Y_{010} - Y_{101})(Z_{011} - Z_{110}) + Y_{101}(Z_{100} - Z_{001})$$
$$+ (Y_{001} - Y_{110})(Z_{101} - Z_{011}) + Y_{110}(Z_{010} - Z_{100})]$$
$$+ F_{100}[(Y_{111} - Y_{000})(Z_{101} - Z_{110}) + Y_{000}(Z_{010} - Z_{001})$$
$$+ (Y_{010} - Y_{101})(Z_{110} - Z_{000}) + Y_{101}(Z_{001} - Z_{111})$$
$$+ (Y_{001} - Y_{110})(Z_{000} - Z_{101}) + Y_{110}(Z_{111} - Z_{010})]$$
$$+ F_{010}[(Y_{111} - Y_{000})(Z_{110} - Z_{011}) + Y_{000}(Z_{001} - Z_{100})$$
$$+ (Y_{100} - Y_{011})(Z_{000} - Z_{110}) + Y_{011}(Z_{111} - Z_{001})$$
$$+ (Y_{001} - Y_{110})(Z_{011} - Z_{000}) + Y_{110}(Z_{100} - Z_{111})]$$
$$+ F_{001}[(Y_{111} - Y_{000})(Z_{011} - Z_{101}) + Y_{000}(Z_{100} - Z_{010})$$
$$+ (Y_{100} - Y_{011})(Z_{101} - Z_{000}) + Y_{011}(Z_{010} - Z_{111})$$
$$+ (Y_{010} - Y_{101})(Z_{000} - Z_{011}) + Y_{101}(Z_{111} - Z_{100})]$$
$$+ F_{000}[(Y_{011} - Y_{100})(Z_{010} - Z_{001}) + Y_{100}(Z_{101} - Z_{110})$$
$$+ (Y_{101} - Y_{010})(Z_{001} - Z_{100}) + Y_{010}(Z_{110} - Z_{011})$$
$$+ (Y_{110} - Y_{001})(Z_{100} - Z_{010}) + Y_{001}(Z_{011} - Z_{101})]$$
$$+ F_{011}[(Y_{000} - Y_{111})(Z_{001} - Z_{010}) + Y_{111}(Z_{110} - Z_{101})$$
$$+ (Y_{101} - Y_{010})(Z_{111} - Z_{001}) + Y_{010}(Z_{000} - Z_{110})$$
$$+ (Y_{110} - Y_{001})(Z_{010} - Z_{111}) + Y_{001}(Z_{101} - Z_{000})]$$
$$+ F_{101}[(Y_{000} - Y_{111})(Z_{100} - Z_{001}) + Y_{111}(Z_{011} - Z_{110})$$
$$+ (Y_{011} - Y_{100})(Z_{001} - Z_{111}) + Y_{100}(Z_{110} - Z_{000})$$
$$+ (Y_{110} - Y_{001})(Z_{111} - Z_{100}) + Y_{001}(Z_{000} - Z_{011})]$$
$$+ F_{110}[(Y_{000} - Y_{111})(Z_{010} - Z_{100}) + Y_{111}(Z_{101} - Z_{011})$$
$$+ (Y_{011} - Y_{100})(Z_{111} - Z_{010}) + Y_{100}(Z_{000} - Z_{101})$$
$$+ (Y_{101} - Y_{010})(Z_{100} - Z_{111}) + Y_{010}(Z_{011} - Z_{000})]\}. \quad (A1)$$

Exactly integrated divergence, *x* component:

$$(\mathcal{D}_x \mathbf{W})_{111} = \frac{1}{12}$$

$$\{\mathbf{W}_{111}[(Y_{211} - Y_{122})(Z_{121} - Z_{112}) + Y_{211}(Z_{221} - Z_{212})$$
$$+ (Y_{121} - Y_{212})(Z_{112} - Z_{211}) + Y_{121}(Z_{122} - Z_{221})$$
$$+ (Y_{112} - Y_{221})(Z_{211} - Z_{121}) + Y_{112}(Z_{212} - Z_{122})]$$
$$+ \mathbf{W}_{100}[(Y_{211} - Y_{100})(Z_{101} - Z_{110}) + Y_{211}(Z_{201} - Z_{210})$$
$$+ (Y_{101} - Y_{210})(Z_{110} - Z_{211}) + Y_{101}(Z_{100} - Z_{201})$$
$$+ (Y_{110} - Y_{201})(Z_{211} - Z_{101}) + Y_{110}(Z_{210} - Z_{100})]$$

$$
\begin{aligned}
+\mathbf{W}_{010}[&(Y_{121} - Y_{010})(Z_{110} - Z_{011}) + Y_{121}(Z_{120} - Z_{021}) \\
&+ (Y_{110} - Y_{021})(Z_{011} - Z_{121}) + Y_{110}(Z_{010} - Z_{120}) \\
&+ (Y_{011} - Y_{120})(Z_{121} - Z_{110}) + Y_{011}(Z_{021} - Z_{010})] \\
+\mathbf{W}_{001}[&(Y_{112} - Y_{001})(Z_{011} - Z_{101}) + Y_{112}(Z_{012} - Z_{102}) \\
&+ (Y_{011} - Y_{102})(Z_{101} - Z_{112}) + Y_{011}(Z_{001} - Z_{012}) \\
&+ (Y_{101} - Y_{012})(Z_{112} - Z_{011}) + Y_{101}(Z_{102} - Z_{001})] \\
+\mathbf{W}_{000}[&(Y_{011} - Y_{100})(Z_{110} - Z_{101}) + Y_{011}(Z_{010} - Z_{001}) \\
&+ (Y_{101} - Y_{010})(Z_{011} - Z_{110}) + Y_{101}(Z_{001} - Z_{100}) \\
&+ (Y_{110} - Y_{001})(Z_{101} - Z_{011}) + Y_{110}(Z_{100} - Z_{010})] \\
+\mathbf{W}_{011}[&(Y_{011} - Y_{122})(Z_{112} - Z_{121}) + Y_{011}(Z_{012} - Z_{021}) \\
&+ (Y_{121} - Y_{012})(Z_{011} - Z_{112}) + Y_{121}(Z_{021} - Z_{122}) \\
&+ (Y_{112} - Y_{021})(Z_{121} - Z_{011}) + Y_{112}(Z_{122} - Z_{012})] \\
+\mathbf{W}_{101}[&(Y_{101} - Y_{212})(Z_{211} - Z_{112}) + Y_{101}(Z_{201} - Z_{102}) \\
&+ (Y_{112} - Y_{201})(Z_{101} - Z_{211}) + Y_{112}(Z_{102} - Z_{212}) \\
&+ (Y_{211} - Y_{102})(Z_{112} - Z_{101}) + Y_{211}(Z_{212} - Z_{201})] \\
+\mathbf{W}_{110}[&(Y_{110} - Y_{221})(Z_{121} - Z_{211}) + Y_{110}(Z_{120} - Z_{210}) \\
&+ (Y_{211} - Y_{120})(Z_{110} - Z_{121}) + Y_{211}(Z_{210} - Z_{221}) \\
&+ (Y_{121} - Y_{210})(Z_{211} - Z_{110}) + Y_{121}(Z_{221} - Z_{120})]\}. \quad \text{(A2)}
\end{aligned}
$$

Exactly integrated gradient, *y* component:

$$
(D_y F)_{000} = \frac{1}{12}
$$

$$
\begin{aligned}
\{F_{111}[&(Z_{100} - Z_{011})(X_{110} - X_{101}) + Z_{011}(X_{001} - X_{010}) \\
&+ (Z_{010} - Z_{101})(X_{011} - X_{110}) + Z_{101}(X_{100} - X_{001}) \\
&+ (Z_{001} - Z_{110})(X_{101} - X_{011}) + Z_{110}(X_{010} - X_{100})] \\
+F_{100}[&(Z_{111} - Z_{000})(X_{101} - X_{110}) + Z_{000}(X_{010} - X_{001}) \\
&+ (Z_{010} - Z_{101})(X_{110} - X_{000}) + Z_{101}(X_{001} - X_{111}) \\
&+ (Z_{001} - Z_{110})(X_{000} - X_{101}) + Z_{110}(X_{111} - X_{010})] \\
+F_{010}[&(Z_{111} - Z_{000})(X_{110} - X_{011}) + Z_{000}(X_{001} - X_{100}) \\
&+ (Z_{100} - Z_{011})(X_{000} - X_{110}) + Z_{011}(X_{111} - X_{001}) \\
&+ (Z_{001} - Z_{110})(X_{011} - X_{000}) + Z_{110}(X_{100} - X_{111})] \\
+F_{001}[&(Z_{111} - Z_{000})(X_{011} - X_{101}) + Z_{000}(X_{100} - X_{010}) \\
&+ (Z_{100} - Z_{011})(X_{101} - X_{000}) + Z_{011}(X_{010} - X_{111}) \\
&+ (Z_{010} - Z_{101})(X_{000} - X_{011}) + Z_{101}(X_{111} - X_{100})] \\
+F_{000}[&(Z_{011} - Z_{100})(X_{010} - X_{001}) + Z_{100}(X_{101} - X_{110}) \\
&+ (Z_{101} - Z_{010})(X_{001} - X_{100}) + Z_{010}(X_{110} - X_{011}) \\
&+ (Z_{110} - Z_{001})(X_{100} - X_{010}) + Z_{001}(X_{011} - X_{101})] \\
+F_{011}[&(Z_{000} - Z_{111})(X_{001} - X_{010}) + Z_{111}(X_{110} - X_{101}) \\
&+ (Z_{101} - Z_{010})(X_{111} - X_{001}) + Z_{010}(X_{000} - X_{110}) \\
&+ (Z_{110} - Z_{001})(X_{010} - X_{111}) + Z_{001}(X_{101} - X_{000})] \\
+F_{101}[&(Z_{000} - Z_{111})(X_{100} - X_{001}) + Z_{111}(X_{011} - X_{110}) \\
&+ (Z_{011} - Z_{100})(X_{001} - X_{111}) + Z_{100}(X_{110} - X_{000}) \\
&+ (Z_{110} - Z_{001})(X_{111} - X_{100}) + Z_{001}(X_{000} - X_{011})] \\
+F_{110}[&(Z_{000} - Z_{111})(X_{010} - X_{100}) + Z_{111}(X_{101} - X_{011}) \\
&+ (Z_{011} - Z_{100})(X_{111} - X_{010}) + Z_{100}(X_{000} - X_{101}) \\
&+ (Z_{101} - Z_{010})(X_{100} - X_{111}) + Z_{010}(X_{011} - X_{000})]\}. \quad \text{(A3)}
\end{aligned}
$$

Exactly integrated divergence, *y* component:

$$
(\mathcal{D}_y \mathbf{W})_{111} = \frac{1}{12}
$$

$$
\begin{aligned}
\{\mathbf{W}_{111}[&(Z_{211} - Z_{122})(X_{121} - X_{112}) + Z_{211}(X_{221} - X_{212}) \\
&+ (Z_{121} - Z_{212})(X_{112} - X_{211}) + Z_{121}(X_{122} - X_{221}) \\
&+ (Z_{112} - Z_{221})(X_{211} - X_{121}) + Z_{112}(X_{212} - X_{122})] \\
+\mathbf{W}_{100}[&(Z_{211} - Z_{100})(X_{101} - X_{110}) + Z_{211}(X_{201} - X_{210}) \\
&+ (Z_{101} - Z_{210})(X_{110} - X_{211}) + Z_{101}(X_{100} - X_{201}) \\
&+ (Z_{110} - Z_{201})(X_{211} - X_{101}) + Z_{110}(X_{210} - X_{100})] \\
+\mathbf{W}_{010}[&(Z_{121} - Z_{010})(X_{110} - X_{011}) + Z_{121}(X_{120} - X_{021}) \\
&+ (Z_{110} - Z_{021})(X_{011} - X_{121}) + Z_{110}(X_{010} - X_{120}) \\
&+ (Z_{011} - Z_{120})(X_{121} - X_{110}) + Z_{011}(X_{021} - X_{010})] \\
+\mathbf{W}_{001}[&(Z_{112} - Z_{001})(X_{011} - X_{101}) + Z_{112}(X_{012} - X_{102}) \\
&+ (Z_{011} - Z_{102})(X_{101} - X_{112}) + Z_{011}(X_{001} - X_{012}) \\
&+ (Z_{101} - Z_{012})(X_{112} - X_{011}) + Z_{101}(X_{102} - X_{001})] \\
+\mathbf{W}_{000}[&(Z_{011} - Z_{100})(X_{110} - X_{101}) + Z_{011}(X_{010} - X_{001}) \\
&+ (Z_{101} - Z_{010})(X_{011} - X_{110}) + Z_{101}(X_{001} - X_{100}) \\
&+ (Z_{110} - Z_{001})(X_{101} - X_{011}) + Z_{110}(X_{100} - X_{010})] \\
+\mathbf{W}_{011}[&(Z_{011} - Z_{122})(X_{112} - X_{121}) + Z_{011}(X_{012} - X_{021}) \\
&+ (Z_{121} - Z_{012})(X_{011} - X_{112}) + Z_{121}(X_{021} - X_{122}) \\
&+ (Z_{112} - Z_{021})(X_{121} - X_{011}) + Z_{112}(X_{122} - X_{012})] \\
+\mathbf{W}_{101}[&(Z_{101} - Z_{212})(X_{211} - X_{112}) + Z_{101}(X_{201} - X_{102}) \\
&+ (Z_{112} - Z_{201})(X_{101} - X_{211}) + Z_{112}(X_{102} - X_{212}) \\
&+ (Z_{211} - Z_{102})(X_{112} - X_{101}) + Z_{211}(X_{212} - X_{201})] \\
+\mathbf{W}_{110}[&(Z_{110} - Z_{221})(X_{121} - X_{211}) + Z_{110}(X_{120} - X_{210}) \\
&+ (Z_{211} - Z_{120})(X_{110} - X_{121}) + Z_{211}(X_{210} - X_{221}) \\
&+ (Z_{121} - Z_{210})(X_{211} - X_{110}) + Z_{121}(X_{221} - X_{120})]\}. \quad \text{(A4)}
\end{aligned}
$$

Exactly integrated gradient, *z* component:

$$
(D_z F)_{000} = \frac{1}{12}
$$

$$
\begin{aligned}
\{F_{111}[&(X_{100} - X_{011})(Y_{110} - Y_{101}) + X_{011}(Y_{001} - Y_{010}) \\
&+ (X_{010} - X_{101})(Y_{011} - Y_{110}) + X_{101}(Y_{100} - Y_{001}) \\
&+ (X_{001} - X_{110})(Y_{101} - Y_{011}) + X_{110}(Y_{010} - Y_{100})] \\
+F_{100}[&(X_{111} - X_{000})(Y_{101} - Y_{110}) + X_{000}(Y_{010} - Y_{001}) \\
&+ (X_{010} - X_{101})(Y_{110} - Y_{000}) + X_{101}(Y_{001} - Y_{111}) \\
&+ (X_{001} - X_{110})(Y_{000} - Y_{101}) + X_{110}(Y_{111} - Y_{010})] \\
+F_{010}[&(X_{111} - X_{000})(Y_{110} - Y_{011}) + X_{000}(Y_{001} - Y_{100}) \\
&+ (X_{100} - X_{011})(Y_{000} - Y_{110}) + X_{011}(Y_{111} - Y_{001}) \\
&+ (X_{001} - X_{110})(Y_{011} - Y_{000}) + X_{110}(Y_{100} - Y_{111})] \\
+F_{001}[&(X_{111} - X_{000})(Y_{011} - Y_{101}) + X_{000}(Y_{100} - Y_{010}) \\
&+ (X_{100} - X_{011})(Y_{101} - Y_{000}) + X_{011}(Y_{010} - Y_{111}) \\
&+ (X_{010} - X_{101})(Y_{000} - Y_{011}) + X_{101}(Y_{111} - Y_{100})] \\
+F_{000}[&(X_{011} - X_{100})(Y_{010} - Y_{001}) + X_{100}(Y_{101} - Y_{110}) \\
&+ (X_{101} - X_{010})(Y_{001} - Y_{100}) + X_{010}(Y_{110} - Y_{011}) \\
&+ (X_{110} - X_{001})(Y_{100} - Y_{010}) + X_{001}(Y_{011} - Y_{101})]
\end{aligned}
$$

$+F_{011}[(X_{000} - X_{111})(Y_{001} - Y_{010}) + X_{111}(Y_{110} - Y_{101})$

$\quad + (X_{101} - X_{010})(Y_{111} - Y_{001}) + X_{010}(Y_{000} - Y_{110})$

$\quad + (X_{110} - X_{001})(Y_{010} - Y_{111}) + X_{001}(Y_{101} - Y_{000})]$

$+F_{101}[(X_{000} - X_{111})(Y_{100} - Y_{001}) + X_{111}(Y_{011} - Y_{110})$

$\quad + (X_{011} - X_{100})(Y_{001} - Y_{111}) + X_{100}(Y_{110} - Y_{000})$

$\quad + (X_{110} - X_{001})(Y_{111} - Y_{100}) + X_{001}(Y_{000} - Y_{011})]$

$+F_{110}[(X_{000} - X_{111})(Y_{010} - Y_{100}) + X_{111}(Y_{101} - Y_{011})$

$\quad + (X_{011} - X_{100})(Y_{111} - Y_{010}) + X_{100}(Y_{000} - Y_{101})$

$\quad + (X_{101} - X_{010})(Y_{100} - Y_{111}) + X_{010}(Y_{011} - Y_{000})]\}.$ (A5)

Exactly integrated divergence, z component:

$$(\mathcal{D}_z\mathbf{W})_{111} = \frac{1}{12}$$

$\{\mathbf{W}_{111}[(X_{211} - X_{122})(Y_{121} - Y_{112}) + X_{211}(Y_{221} - Y_{212})$

$\quad + (X_{121} - X_{212})(Y_{112} - Y_{211}) + X_{121}(Y_{122} - Y_{221})$

$\quad + (X_{112} - X_{221})(Y_{211} - Y_{121}) + X_{112}(Y_{212} - Y_{122})]$

$+\mathbf{W}_{100}[(X_{211} - X_{100})(Y_{101} - Y_{110}) + X_{211}(Y_{201} - Y_{210})$

$\quad + (X_{101} - X_{210})(Y_{110} - Y_{211}) + X_{101}(Y_{100} - Y_{201})$

$\quad + (X_{110} - X_{201})(Y_{211} - Y_{101}) + X_{110}(Y_{210} - Y_{100})]$

$+\mathbf{W}_{010}[(X_{121} - X_{010})(Y_{110} - Y_{011}) + X_{121}(Y_{120} - Y_{021})$

$\quad + (X_{110} - X_{021})(Y_{011} - Y_{121}) + X_{110}(Y_{010} - Y_{120})$

$\quad + (X_{011} - X_{120})(Y_{121} - Y_{110}) + X_{011}(Y_{021} - Y_{010})]$

$+\mathbf{W}_{001}[(X_{112} - X_{001})(Y_{011} - Y_{101}) + X_{112}(Y_{012} - Y_{102})$

$\quad + (X_{011} - X_{102})(Y_{101} - Y_{112}) + X_{011}(Y_{001} - Y_{012})$

$\quad + (X_{101} - X_{012})(Y_{112} - Y_{011}) + X_{101}(Y_{102} - Y_{001})]$

$+\mathbf{W}_{000}[(X_{011} - X_{100})(Y_{110} - Y_{101}) + X_{011}(Y_{010} - Y_{001})$

$\quad + (X_{101} - X_{010})(Y_{011} - Y_{110}) + X_{101}(Y_{001} - Y_{100})$

$\quad + (X_{110} - X_{001})(Y_{101} - Y_{011}) + X_{110}(Y_{100} - Y_{010})]$

$+\mathbf{W}_{011}[(X_{011} - X_{122})(Y_{112} - Y_{121}) + X_{011}(Y_{012} - Y_{021})$

$\quad + (X_{121} - X_{012})(Y_{011} - Y_{112}) + X_{121}(Y_{021} - Y_{122})$

$\quad + (X_{112} - X_{021})(Y_{121} - Y_{011}) + X_{112}(Y_{122} - Y_{012})]$

$+\mathbf{W}_{101}[(X_{101} - X_{212})(Y_{211} - Y_{112}) + X_{101}(Y_{201} - Y_{102})$

$\quad + (X_{112} - X_{201})(Y_{101} - Y_{211}) + X_{112}(Y_{102} - Y_{212})$

$\quad + (X_{211} - X_{102})(Y_{112} - Y_{101}) + X_{211}(Y_{212} - Y_{201})]$

$+\mathbf{W}_{110}[(X_{110} - X_{221})(Y_{121} - Y_{211}) + X_{110}(Y_{120} - Y_{210})$

$\quad + (X_{211} - X_{120})(Y_{110} - Y_{121}) + X_{211}(Y_{210} - Y_{221})$

$\quad + (X_{121} - X_{210})(Y_{211} - Y_{110}) + X_{121}(Y_{221} - Y_{120})]\}.$ (A6)

One-point quadrature gradient, x component:

$$(D_x F)_{000} = \frac{1}{16}$$

$\{(F_{111} - F_{000})[(Y_{100} - Y_{011})(Z_{010} - Z_{101} - Z_{001} + Z_{110})$

$\quad + (Y_{010} - Y_{101})(Z_{001} - Z_{110} - Z_{100} + Z_{011})$

$\quad + (Y_{001} - Y_{110})(Z_{100} - Z_{011} - Z_{010} + Z_{101})]$

$+(F_{100} - F_{011})[(Y_{111} - Y_{000})(Z_{001} - Z_{110} - Z_{010} + Z_{101})$

$\quad + (Y_{010} - Y_{101})(Z_{111} - Z_{000} - Z_{001} + Z_{110})$

$\quad + (Y_{001} - Y_{110})(Z_{010} - Z_{101} - Z_{111} + Z_{000})]$

$+(F_{010} - F_{101})[(Y_{111} - Y_{000})(Z_{100} - Z_{011} - Z_{001} + Z_{110})$

$\quad + (Y_{001} - Y_{110})(Z_{111} - Z_{000} - Z_{100} + Z_{011})$

$\quad + (Y_{100} - Y_{011})(Z_{001} - Z_{110} - Z_{111} + Z_{000})]$

$+(F_{001} - F_{110})[(Y_{111} - Y_{000})(Z_{010} - Z_{101} - Z_{100} + Z_{011})$

$\quad + (Y_{100} - Y_{011})(Z_{111} - Z_{000} - Z_{010} + Z_{101})$

$\quad + (Y_{010} - Y_{101})(Z_{100} - Z_{011} - Z_{111} + Z_{000})]\}.$ (A7)

One-point quadrature divergence, x component:

$$(\mathcal{D}_x\mathbf{W})_{111} = \frac{1}{16}$$

$\{\mathbf{W}_{111}[(Y_{211} - Y_{122})(Z_{121} - Z_{212} - Z_{112} + Z_{221})$

$\quad + (Y_{121} - Y_{212})(Z_{112} - Z_{221} - Z_{211} + Z_{122})$

$\quad + (Y_{112} - Y_{221})(Z_{211} - Z_{122} - Z_{121} + Z_{212})]$

$+\mathbf{W}_{100}[(Y_{211} - Y_{100})(Z_{101} - Z_{210} - Z_{110} + Z_{201})$

$\quad + (Y_{101} - Y_{210})(Z_{110} - Z_{201} - Z_{211} + Z_{100})$

$\quad + (Y_{110} - Y_{201})(Z_{211} - Z_{100} - Z_{101} + Z_{210})]$

$+\mathbf{W}_{010}[(Y_{121} - Y_{010})(Z_{110} - Z_{021} - Z_{011} + Z_{120})\}.$ (A8)

One-point quadrature gradient, y component:

$$(D_y F)_{000} = \frac{1}{16}$$

$\{(F_{111} - F_{000})[(Z_{100} - Z_{011})(X_{010} - X_{101} - X_{001} + X_{110})$

$\quad + (Z_{010} - Z_{101})(X_{001} - X_{110} - X_{100} + X_{011})$

$\quad + (Z_{001} - Z_{110})(X_{100} - X_{011} - X_{010} + X_{101})]$

$+(F_{100} - F_{011})[(Z_{111} - Z_{000})(X_{001} - X_{110} - X_{010} + X_{101})$

$\quad + (Z_{010} - Z_{101})(X_{111} - X_{000} - X_{001} + X_{110})$

$\quad + (Z_{001} - Z_{110})(X_{010} - X_{101} - X_{111} + X_{000})]$

$+(F_{010} - F_{101})[(Z_{111} - Z_{000})(X_{100} - X_{011} - X_{001} + X_{110})$

$\quad + (Z_{001} - Z_{110})(X_{111} - X_{000} - X_{100} + X_{011})$

$\quad + (Z_{100} - Z_{011})(X_{001} - X_{110} - X_{111} + X_{000})]$

$+(F_{001} - F_{110})[(Z_{111} - Z_{000})(X_{010} - X_{101} - X_{100} + X_{011})$

$\quad + (Z_{100} - Z_{011})(X_{111} - X_{000} - X_{010} + X_{101})$

$\quad + (Z_{010} - Z_{101})(X_{100} - X_{011} - X_{111} + X_{000})]\}.$ (A9)

One-point quadrature divergence, y component:

$$(\mathcal{D}_y\mathbf{W})_{111} = \tfrac{1}{16}$$

$\{\mathbf{W}_{111}[(Z_{211} - Z_{122})(X_{121} - X_{212} - X_{112} + X_{221})$

$\quad + (Z_{121} - Z_{212})(X_{112} - X_{221} - X_{211} + X_{122})$

$\quad + (Z_{112} - Z_{221})(X_{211} - X_{122} - X_{121} + X_{212})]$

$+\mathbf{W}_{100}[(Z_{211} - Z_{100})(X_{101} - X_{210} - X_{110} + X_{201})$

$\quad + (Z_{101} - Z_{210})(X_{110} - X_{201} - X_{211} + X_{100})$

$\quad + (Z_{110} - Z_{201})(X_{211} - X_{100} - X_{101} + X_{210})]$

$+\mathbf{W}_{010}[(Z_{121} - Z_{010})(X_{110} - X_{021} - X_{011} + X_{120})$

$\quad + (Z_{110} - Z_{021})(X_{011} - X_{120} - X_{121} + X_{010})$

$\quad + (Z_{011} - Z_{120})(X_{121} - X_{010} - X_{110} + X_{021})]$

$+\mathbf{W}_{001}[(Z_{112} - Z_{001})(X_{011} - X_{102} - X_{101} + X_{012})$

$\quad + (Z_{011} - Z_{102})(X_{101} - X_{012} - X_{112} + X_{001})$

$\quad + (Z_{101} - Z_{012})(X_{112} - X_{001} - X_{011} + X_{102})]$

$+\mathbf{W}_{000}[(Z_{011} - Z_{100})(X_{010} - X_{101} - X_{001} + X_{110})$

$\quad + (Z_{101} - Z_{010})(X_{001} - X_{110} - X_{100} + X_{011})$

$\quad + (Z_{110} - Z_{001})(X_{100} - X_{011} - X_{010} + X_{101})]$

$+\mathbf{W}_{011}[(Z_{011} - Z_{122})(X_{012} - X_{121} - X_{021} + X_{112})$

$\quad + (Z_{121} - Z_{012})(X_{021} - X_{112} - X_{122} + X_{011})$

$\quad + (Z_{112} - Z_{021})(X_{122} - X_{011} - X_{012} + X_{121})]$

$+\mathbf{W}_{101}[(Z_{101} - Z_{212})(X_{201} - X_{112} - X_{102} + X_{211})$

$\quad + (Z_{112} - Z_{201})(X_{102} - X_{211} - X_{212} + X_{101})$

$\quad + (Z_{211} - Z_{102})(X_{212} - X_{101} - X_{201} + X_{112})]$

$+\mathbf{W}_{110}[(Z_{110} - Z_{221})(X_{120} - X_{211} - X_{210} + X_{121})$

$\quad + (Z_{211} - Z_{120})(X_{210} - X_{121} - X_{221} + X_{110})$

$\quad + (Z_{121} - Z_{210})(X_{221} - X_{110} - X_{120} + X_{211})]\}. \qquad \text{(A10)}$

One-point quadrature gradient, *z* component:

$(D_z F)_{000} = \dfrac{1}{16}$

$\{(F_{111} - F_{000})[(X_{100} - X_{011})(Y_{010} - Y_{101} - Y_{001} + Y_{110})$

$\quad + (X_{010} - X_{101})(Y_{001} - Y_{110} - Y_{100} + Y_{011})$

$\quad + (X_{001} - X_{110})(Y_{100} - Y_{011} - Y_{010} + Y_{101})]$

$+(F_{100} - F_{011})[(X_{111} - X_{000})(Y_{001} - Y_{110} - Y_{010} + Y_{101})$

$\quad + (X_{010} - X_{101})(Y_{111} - Y_{000} - Y_{001} + Y_{110})$

$\quad + (X_{001} - X_{110})(Y_{010} - Y_{101} - Y_{111} + Y_{000})]$

$+(F_{010} - F_{101})[(X_{111} - X_{000})(Y_{100} - Y_{011} - Y_{001} + Y_{110})$

$\quad + (X_{001} - X_{110})(Y_{111} - Y_{000} - Y_{100} + Y_{011})$

$\quad + (X_{100} - X_{011})(Y_{001} - Y_{110} - Y_{111} + Y_{000})]$

$+(F_{001} - F_{110})[(X_{111} - X_{000})(Y_{010} - Y_{101} - Y_{100} + Y_{011})$

$\quad + (X_{100} - X_{011})(Y_{111} - Y_{000} - Y_{010} + Y_{101})$

$\quad + (X_{010} - X_{101})(Y_{100} - Y_{011} - Y_{111} + Y_{000})]\}. \qquad \text{(A11)}$

One-point quadrature divergence, *z* component:

$(\mathcal{D}_z \mathbf{W})_{111} = \dfrac{1}{16}$

$\{\mathbf{W}_{111}[(X_{211} - X_{122})(Y_{121} - Y_{212} - Y_{112} + Y_{221})$

$\quad + (X_{121} - X_{212})(Y_{112} - Y_{221} - Y_{211} + Y_{122})$

$\quad + (X_{112} - X_{221})(Y_{211} - Y_{122} - Y_{121} + Y_{212})]$

$+\mathbf{W}_{100}[(X_{211} - X_{100})(Y_{101} - Y_{210} - Y_{110} + Y_{201})$

$\quad + (X_{101} - X_{210})(Y_{110} - Y_{201} - Y_{211} + Y_{100})$

$\quad + (X_{110} - X_{201})(Y_{211} - Y_{100} - Y_{101} + Y_{210})]$

$+\mathbf{W}_{010}[(X_{121} - X_{010})(Y_{110} - Y_{021} - Y_{011} + Y_{120})$

$\quad + (X_{110} - X_{021})(Y_{011} - Y_{120} - Y_{121} + Y_{010})$

$\quad + (X_{011} - X_{120})(Y_{121} - Y_{010} - Y_{110} + Y_{021})]$

$+\mathbf{W}_{001}[(X_{112} - X_{001})(Y_{011} - Y_{102} - Y_{101} + Y_{012})$

$\quad + (X_{011} - X_{102})(Y_{101} - Y_{012} - Y_{112} + Y_{001})$

$\quad + (X_{101} - X_{012})(Y_{112} - Y_{001} - Y_{011} + Y_{102})]$

$+\mathbf{W}_{000}[(X_{011} - X_{100})(Y_{010} - Y_{101} - Y_{001} + Y_{110})$

$\quad + (X_{101} - X_{010})(Y_{001} - Y_{110} - Y_{100} + Y_{011})$

$\quad + (X_{110} - X_{001})(Y_{100} - Y_{011} - Y_{010} + Y_{101})]$

$+\mathbf{W}_{011}[(X_{011} - X_{122})(Y_{012} - Y_{121} - Y_{021} + Y_{112})$

$\quad + (X_{121} - X_{012})(Y_{021} - Y_{112} - Y_{122} + Y_{011})$

$\quad + (X_{112} - X_{021})(Y_{122} - Y_{011} - Y_{012} + Y_{121})]$

$+\mathbf{W}_{101}[(X_{101} - X_{212})(Y_{201} - Y_{112} - Y_{102} + Y_{211})$

$\quad + (X_{112} - X_{201})(Y_{102} - Y_{211} - Y_{212} + Y_{101})$

$\quad + (X_{211} - X_{102})(Y_{212} - Y_{101} - Y_{201} + Y_{112})]$

$+\mathbf{W}_{110}[(X_{110} - X_{221})(Y_{120} - Y_{211} - Y_{210} + Y_{121})$

$\quad + (X_{211} - X_{120})(Y_{210} - Y_{121} - Y_{221} + Y_{110})$

$\quad + (X_{121} - X_{210})(Y_{221} - Y_{110} - Y_{120} + Y_{211})]\}. \qquad \text{(A12)}$