Geophys. J. Int. (1999) **139,** 273–282

Interpolation of ray theory traveltimes within ray cells

Petr Bulant and Luděk Klimeš

Department of Geophysics, Charles University, Ke Karlovu 3, 121 16 *Praha* 2, *Czech Republic. E-mail: bulant@seis.karlov.mff.cuni.cz; klimes@seis.karlov.mff.cuni.cz*

Accepted 1999 May 11. Received 1999 April 23; in original form 1998 December 15

SUMMARY

3-D ray tracing followed by interpolation of the computed quantities amongst the rays is a powerful tool for the computation of ray theory traveltimes, amplitudes and other quantities at the gridpoints of dense rectangular grids. Several methods based on the decomposition of the model volume into ray cells, and on further interpolation within the individual ray cells, have recently been introduced. We propose bilinear and bicubic interpolation schemes, which should be applicable in any modelling method based on interpolation within ray cells. The bicubic interpolation is designed for traveltimes and the bilinear interpolation for other quantities.

The interpolation schemes have been incorporated into a controlled initial-value ray tracing program package. Multivalued ray theory traveltime computation in a heterogeneous 3-D model is shown as an example.

Key words: inhomogeneous media, ray theory, ray tracing, seismic modelling, wave propagation, traveltime.

1 INTRODUCTION

3-D ray tracing, especially the computation of ray theory traveltimes at nodes of dense grids of points, has become quite important in recent years. The possibility of computing traveltimes is of principal importance in the imaging of depth structures, seismic tomography and hypocentre determination techniques. Several methods based on decomposition of the model volume into ray cells, and on further interpolation within individual ray cells were introduced.

In the *wave front construction* method (Vinje *et al.* 1996a,b) the entire wave front is represented by a triangular mesh with a ray at each node. The wave front is propagated time-step by time-step through the model. The space between two consecutive wave fronts consists of ray cells, each of them being bounded by a triangle at the older wave front, by the corresponding triangle at the newer wave front, and by segments of the three rays connecting the triangles. The receivers located within the ray cells are assigned values of interpolated traveltimes and other quantities. If the difference between neighbouring rays on the newer wave front exceeds some pre-defined level, new rays are interpolated on the older wave front, splitting ray cells into smaller parts.

In the *recursive seismic ray modelling* method of Moser & Pajchel (1997), the computation is organized as a 'depth-first search', rather than a 'breadth-first search' in the wave front construction method. The computation of a selected ray cell is followed by the computation of the cell in the next time step. At each interface, ray tubes corresponding to all the elementary waves under consideration are recursively generated. This procedure is continued until the whole ray tube, composed of individual ray cells, is computed, including reflections, transmissions and diffractions. From the point of view of interpolation, the main difference as compared to the wave front construction method is that the neighbouring ray cells are not available when interpolating in a ray cell.

In the *controlled initial-value ray tracing* method (Bulant 1997a,b) just ray tubes corresponding to a prescribed elementary wave are considered, without splitting at structural interfaces. The model volume is decomposed into ray tubes in the first step, and the interpolation within the ray cells is carried out in the second step. This is because the method was proposed as a post-processing method for a two-point ray tracing algorithm based on the triangulation of the domain of ray take-off parameters (for details, see Bulant 1997a,b). Thus, the system of ray tubes and ray cells is given before the interpolation process takes place. Each ray cell corresponds to the space in the ray tube, limited by two planes which approximate wave fronts or structural interfaces. The values of the interpolated quantities at the vertices of a ray cell are available for interpolation.

Whereas the above-mentioned methods differ in the manner of the decomposition of the model volume into ray cells, they all contain some procedure of interpolation of the computed quantities to the receivers (usually nodes of a rectangular grid) located within a single ray cell.

Lucio *et al.* (1996) divided each ray cell into three tetrahedra and performed linear interpolation inside the tetrahedra.

Vinje *et al.* (1996a,b) parametrized a ray cell by two barycentric coordinates, u, v, at the triangular cell bottom and by traveltime t in the ray-tangential direction of the cell, projected

the receiver to an initial point at the triangular bottom of the ray cell, interpolated the initial parameters of a new ray, and traced the new ray from the initial point to the receiver. Their parametrization is thus very similar to our approach (see below), but their interpolation along the cell bottom, probably also cubic, is not described in detail.

We propose bilinear and bicubic interpolation schemes. The bilinear interpolation is easy and robust. It incorporates both the decision whether a receiver lies in the ray cell and the interpolation of quantities computed along the rays to the receiver. The scheme uses only the values of quantities at the vertices of the ray cell, thus the scheme should be applicable in any modelling method based on interpolation within ray cells. If the partial derivatives are known in addition to the functional values at the vertices of the ray cell, the bicubic interpolation scheme may be used to increase the accuracy of the interpolation. The bicubic interpolation is designed for traveltimes and the bilinear interpolation for other quantities.

2 DECOMPOSITION OF A RAY TUBE INTO RAY CELLS

As mentioned above, first we trace rays and decompose the model volume into ray tubes. Each *ray tube* is bounded by three rays, which form a triangle in a 2-D domain of ray take-off parameters. We triangulate the ray take-off parameter domain by so-called *homogeneous triangles*. Each homogeneous triangle is formed by rays of the same *ray history*, that is, by rays which pass the same model blocks, interact with the same interfaces, display the same behaviour at caustics, and terminate in the same area for the same reason. Thus, the width of the ray tubes may be kept below a prescribed limit, and the possibility of interpolation within the ray tubes is guaranteed (see Fig. 1).

For details on triangulation of the ray take-off parameter domain by homogeneous triangles (that is, details on creating the ray tubes), see Bulant (1996). Note that the demarcation belts between homogeneous triangles of different ray histories produce gaps between the corresponding ray tubes. These gaps may then cause gaps in the interpolated quantities in the receiver grid, where traveltimes and other quantities cannot be interpolated. To reduce the gaps as much as possible, we must decide carefully what information we wish to include in the ray histories.

After being computed, each ray is stored on a disk as a set of points of two types. First, there are the points on the ray stored with traveltime step Δt; we shall refer to these as '*time points*'. Second, there are the points of interaction of the ray with interfaces and other surfaces (reflection, transmission or termination); we shall refer to these as '*interaction points*'. A ray tube is thus represented as a set of the above points on the three rays which form the ray tube (see Fig. 2). For interpolation to be feasible, the ray tube must be decomposed into smaller parts.

We decompose the ray tube into *ray cells*. *Regular ray cells* are defined by six points on the rays. Both the bottom and the top of the regular ray cell are formed by triangles defined by three points on the rays. Mostly these are the 'time points' of the same traveltime level t. *Degenerate ray cells* are formed by five or four points. They occur mainly at a point source, where all three points of the bottom of a ray cell coincide, before and after structural interfaces, where one or two points of the top of a ray cell coincide with the corresponding point(s) of the bottom, and before the end surface where the ray tube terminates.

When decomposing ray tubes into ray cells, we must ensure that the sides of all the ray cells are the same for neighbouring ray tubes. Otherwise, some gridpoints of a receiver grid might be considered to be located within none or within both of the neighbouring ray cells.

We start the decomposition at the source, taking the first points on the rays as the bottom of the first ray cell, thus the bottom is formed by a single point for point sources, by a line segment for straight-line sources, and by a triangle for curved-line sources and surface sources. We then proceed along the rays to the second points on the rays, taking the second points simultaneously as the top of the first ray cell and the bottom of the second ray cell. We continue this procedure along the rays and create regular ray cells until we reach the first 'interaction

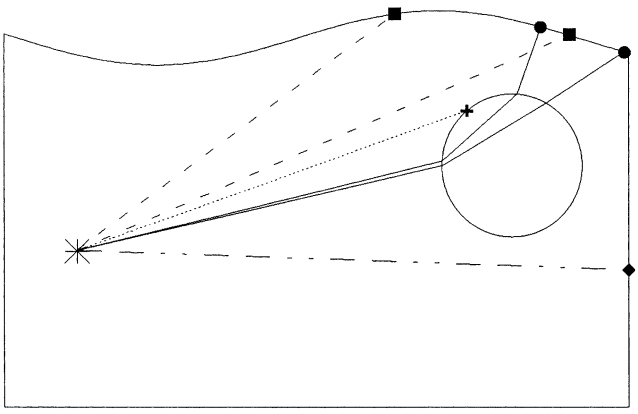

Figure 1. Rays with different ray histories in a seismic model. Only rays with the same ray histories, displayed here by lines ending with the same symbols, are suitable for creating ray tubes to be used to interpolate amongst the rays.

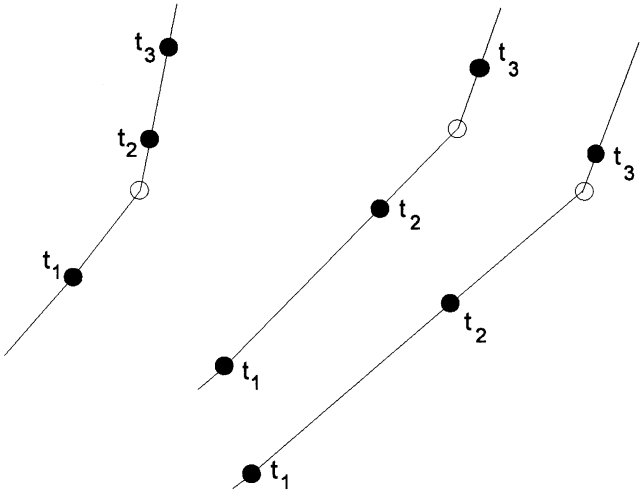

Figure 2. Points on three rays forming a ray tube in the place where the rays cross an interface. Filled circles represent the points at three traveltime levels t_1, t_2 and t_3 ('time points'). Open circles are the points of intersection of the rays with the structural interface ('interaction points').

point' on any of the rays. Here we postpone proceeding along this ray, and we proceed only along the remaining two rays (or one ray) until we reach the 'interaction points' on all three rays. We may thus create one or more degenerate ray cell.

Now the three 'interaction points' form the top of the last ray cell created. If they lie at the end surface, the decomposition of the ray tube into ray cells is completed. If they are located at a structural interface, we continue the process of creating ray cells beyond this interface. We may again have to postpone proceeding along some ray(s) and create some degenerate ray cell(s) until we get 'time points' with the same traveltime at the top of the recently created ray cell. We can then continue with creating regular ray cells with equal traveltimes at their bottoms and tops (see Fig. 3).

Note that once we have reached an 'interaction point' on any one of the three rays, we are certain to reach 'interaction points' of the same kind on the remaining two rays also, because all the rays of the ray tube have the same ray history. It should also be pointed out that the method described above of decomposition of a ray tube into ray cells prevents us from interpolating across structural interfaces.

3 DETERMINATION OF A RAY CELL

By determining a ray cell we understand whether or not an arbitrary point lies within the ray cell. Assume a ray cell defined by six points, $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{B}_3$, $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$, on three rays, where points $\mathbf{B}_i = (B_i^1, B_i^2, B_i^3)^{\mathrm{T}}$ are at the bottom and points $\mathbf{C}_i$ are at the top of the ray cell (see Fig. 4). We introduce a local coordinate system parametrized by parameters w_1, w_2, w_B. The coordinates of an arbitrary point $\mathbf{X}$ are then given as

$$\mathbf{X} = \mathbf{A}_3 + w_1(\mathbf{A}_1 - \mathbf{A}_3) + w_2(\mathbf{A}_2 - \mathbf{A}_3)\,, \tag{1}$$

where

$$\mathbf{A}_i = \mathbf{C}_i + w_B(\mathbf{B}_i - \mathbf{C}_i)\,, \quad i = 1, 2, 3\,. \tag{2}$$

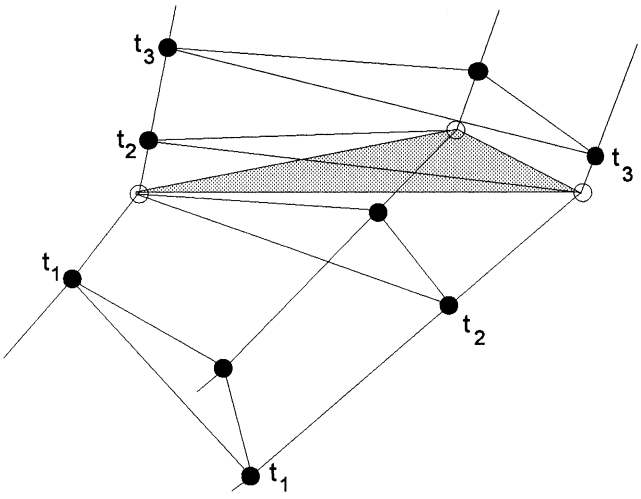

Figure 3. The ray tube from Fig. 2 decomposed into ray cells. There is one regular ray cell formed by six points, then two degenerate cells formed by five and four points, and then a regular ray cell. The bottom of the first ray cell and the top of the fourth cell are formed by triangles which approximate wave fronts at traveltime levels t_1 and t_3. The shaded triangle, which is the top of the second cell and the bottom of the third cell, approximates the structural interface.

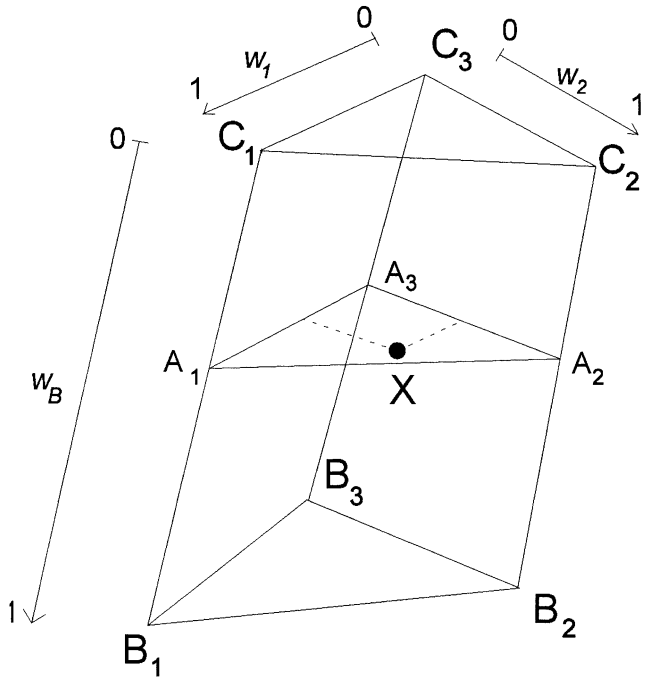

Figure 4. A ray cell is defined by six points $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{B}_3$, $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$, situated on three rays. Point $\mathbf{X}$ is examined to find out whether it is located within the ray cell. Its position may be expressed using the local coordinate system parametrized by parameters w_1, w_2, w_B. If all of the quantities w_1, w_2, w_B, $1-w_1-w_2$ are in the range $\langle 0, 1\rangle$, point $\mathbf{X}$ lies within the ray cell.

Parameters w_1, w_2 are the barycentric coordinates in triangle $\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3$.

Eqs (1) and (2) may also be expressed as

$$\mathbf{X} = w_B \sum_{i=1}^{3} w_i \mathbf{B}_i + w_C \sum_{i=1}^{3} w_i \mathbf{C}_i\,, \tag{3}$$

with $w_C = 1 - w_B$, $\quad w_3 = 1 - w_1 - w_2$. Eq. (1) is the parametric description of planar triangle $\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3$. Eqs (2) are parametric descriptions of the straight-line segments $\mathbf{B}_1\mathbf{C}_1$, $\mathbf{B}_2\mathbf{C}_2$ and $\mathbf{B}_3\mathbf{C}_3$ approximating the ray segments. Note that only in the homogeneous parts of the model do the straight segments lie on the exact rays.

If point $\mathbf{X}$ lies within the ray cell, all values w_1, w_2, w_3, w_B are within the range $\langle 0, 1\rangle$. Thus, for each receiver, which might be located within the selected ray cell, we compute its local coordinates w_1, w_2, w_B and quantities w_3, w_C. If all of them display values in the range $\langle 0, 1\rangle$, point $\mathbf{X}$ is within the ray cell, and the values of all required quantities will be determined by interpolation within the ray cell. Note that only the receivers situated inside the smallest box containing the whole ray cell enter into the computation of the local coordinates.

If the ray cell touches a caustic, two or three (in three dimensions) different sets of values w_1, w_2, w_3, w_B may be found within the range $\langle 0, 1\rangle$ for the same point $\mathbf{X}$ (see Fig. 5). Different coordinates w_1, w_2, w_B of the same point $\mathbf{X}$ correspond to different ray take-off parameters γ_1, γ_2, and the interpolation should be performed for each set of coordinates w_1, w_2, w_B.

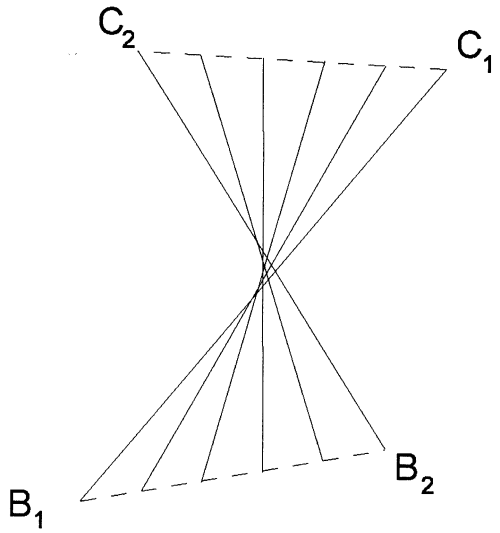

Figure 5. A 2-D ray cell touches a caustic. The lines approximate rays, the dashed lines represent the bottom and the top of the cell. Each point of a small area in the middle of the cell lies on two rays. For such points we get two (two or three in three dimensions) different sets of local coordinates, and then we obtain two (two or three in three dimensions) different values of interpolated quantities.

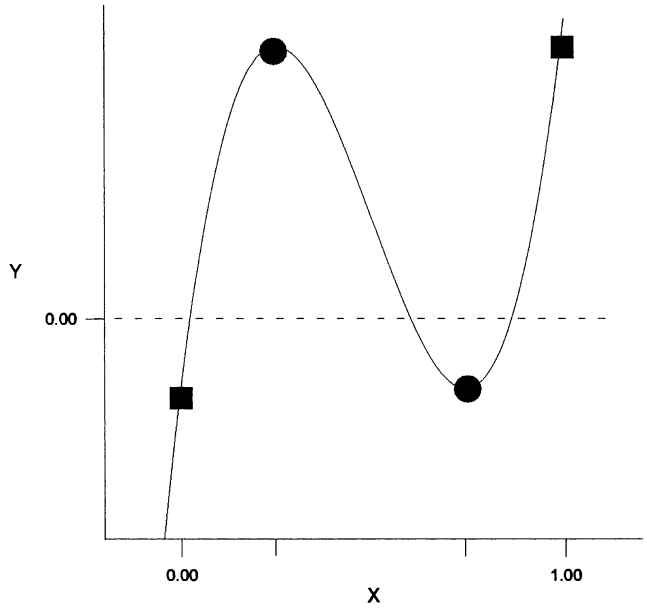

Figure 6. A cubic function with three roots in the range $\langle 0, 1\rangle$. Once we have computed the two stationary points (marked by circles), the interval $\langle 0, 1\rangle$ may be split into three subintervals. Each subinterval contains just one root. The cubic function is monotonic in all the subintervals.

3.1 Computation of w_B

First we compute the value of local coordinate w_B. We know that the volume V of the parallelepiped defined by three arbitrary vectors **M**, **N**, **O** is given by the determinant of the 3×3 matrix $\widehat{\mathbf{M}}$ composed of the three vectors

$$V=\det\widehat{\mathbf{M}}=\begin{vmatrix} M^1 & N^1 & O^1\\ M^2 & N^2 & O^2\\ M^3 & N^3 & O^3\end{vmatrix}=\sum_{i,j,k=1}^{3}\epsilon_{ijk}M^iN^jO^k\,, \tag{4}$$

where ϵ_{ijk} is a Levi–Civita symbol. Point **X** must lie in the plane determined by points $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{A}_3$. Thus, the volume of the parallelepiped defined by vectors $\mathbf{A}_1-\mathbf{X}$, $\mathbf{A}_2-\mathbf{X}$, $\mathbf{A}_3-\mathbf{X}$ should be equal to zero. According to eq. (2),

$$\mathbf{A}_i-\mathbf{X}=\mathbf{C}_i-\mathbf{X}+w_B(\mathbf{B}_i-\mathbf{C}_i)\,,\quad i=1,2,3\,. \tag{5}$$

Considering eqs (4) and (5), we arrive at the cubic equation for parameter w_B in the form

$$\det(\widehat{\mathbf{Z}}+w_B\widehat{\mathbf{Y}})=aw_B^3+bw_B^2+cw_B+d=0\,, \tag{6}$$

where $\widehat{\mathbf{Z}}$ is the 3×3 matrix composed of vectors $\mathbf{C}_i-\mathbf{X}$, $i=1,2,3$, $\widehat{\mathbf{Y}}$ is the 3×3 matrix composed of vectors $\mathbf{B}_i-\mathbf{C}_i$, $i=1,2,3$,

$$a=\det\widehat{\mathbf{Y}}\,, \tag{7}$$

$$b=\sum_{i,j,k=1}^{3}\epsilon_{ijk}Z_1^iY_2^jY_3^k+\sum_{i,j,k=1}^{3}\epsilon_{ijk}Y_1^iZ_2^jY_3^k+\sum_{i,j,k=1}^{3}\epsilon_{ijk}Y_1^iY_2^jZ_3^k\,, \tag{8}$$

$$c=\sum_{i,j,k=1}^{3}\epsilon_{ijk}Y_1^iZ_2^jZ_3^k+\sum_{i,j,k=1}^{3}\epsilon_{ijk}Z_1^iY_2^jZ_3^k+\sum_{i,j,k=1}^{3}\epsilon_{ijk}Z_1^iZ_2^jY_3^k \tag{9}$$

and

$$d=\det\widehat{\mathbf{Z}}\,. \tag{10}$$

Cubic term a equals zero if one of the vectors $\mathbf{B}_i-\mathbf{C}_i$ forming the ray tube is linearly dependent on the other two vectors (including the case in which the vector equals zero). Quadratic term b equals zero if two of the vectors $\mathbf{B}_i-\mathbf{C}_i$ are linearly dependent on the third. Thus for degenerate ray cells, when one or two of points $\mathbf{B}_i$ coincide with the corresponding points $\mathbf{C}_i$, cubic equation (6) reduces to a quadratic or linear equation[1], but the interpolation scheme remains valid. This usually happens when a ray tube interacts with a structural interface (see Fig. 3).

We look for the roots of the cubic equation (6) only in the range $\langle 0, 1\rangle$. Otherwise, the point **X** is out of the ray cell. We tried to solve the equation using analytical formulae from Numerical Recipes in Fortran (Press *et al.* 1996), but the solution was not precise enough for our purposes. Thus we solve the equation numerically. First we compute all the stationary points of the cubic function (6). Then we calculate the values of the cubic function at points 0 and 1 and at all the stationary points from the range $\langle 0, 1\rangle$, if any. We can thus split the interval $\langle 0, 1\rangle$ into subintervals. Each subinterval is limited by points 0, 1, or by the stationary points (see Fig. 6).

We obtain 0–3 subintervals of $\langle 0, 1\rangle$ such that the values of the cubic functions at the endpoints of each subinterval have opposite signs, and each of the subintervals thus contains just

[1] We could thus decompose ray tubes into degenerate ray cells formed by only four points on the rays and take advantage of this important simplification of the cubic equation to a linear equation. This possibility, which is very similar to the approach of Lucio *et al.* (1996), as discussed by Bulant (1999).

one root of cubic equation (6). We can then easily find the roots using the Newton method, for example, because the function is monotonic in all the subintervals. The reader is reminded that all values of w_B from the interval $\langle 0, 1\rangle$ are significant.

3.2 Computation of w_1, w_2

Once we have computed w_B from the interval $\langle 0, 1\rangle$, we proceed to the computation of w_1, w_2, w_3. We calculate the coordinates of points $\mathbf{A}_i$ according to eq. (2). Then, considering eq. (1), we immediately obtain a system of linear equations for w_1, w_2 and w_3,

$$\widehat{\mathbf{A}}\mathbf{W}=\mathbf{X}\,, \tag{11}$$

where $\widehat{\mathbf{A}}$ is a 3×3 matrix composed of vectors $\mathbf{A}_i$, $i=1, 2, 3$, and $\mathbf{W}=(w_1, w_2, w_3)^{\mathrm{T}}$.

However, points $\mathbf{A}_i$ may be very close to one another; vectors $\mathbf{A}_i$ may thus be almost parallel, and the determinant of matrix $\widehat{\mathbf{A}}$ may be very small. This might cause numerical errors in solving eq. (11). It is thus better to construct two perpendicular vectors,

$$\mathbf{E}_1=(\mathbf{A}_1-\mathbf{A}_3)|\mathbf{A}_2-\mathbf{A}_3|+(\mathbf{A}_2-\mathbf{A}_3)|\mathbf{A}_1-\mathbf{A}_3| \tag{12}$$

and

$$\mathbf{E}_2=(\mathbf{A}_1-\mathbf{A}_3)|\mathbf{A}_2-\mathbf{A}_3|-(\mathbf{A}_2-\mathbf{A}_3)|\mathbf{A}_1-\mathbf{A}_3|\,, \tag{13}$$

in the plane defined by points $\mathbf{A}_i$ and then to construct projections,

$$\mathbf{U}_1=((\mathbf{A}_1-\mathbf{A}_3)^{\mathrm{T}}\mathbf{E}_1, (\mathbf{A}_1-\mathbf{A}_3)^{\mathrm{T}}\mathbf{E}_2)^{\mathrm{T}}\,, \tag{14}$$

$$\mathbf{U}_2=((\mathbf{A}_2-\mathbf{A}_3)^{\mathrm{T}}\mathbf{E}_1, (\mathbf{A}_2-\mathbf{A}_3)^{\mathrm{T}}\mathbf{E}_2)^{\mathrm{T}}\,, \tag{15}$$

$$\mathbf{V}=((\mathbf{X}-\mathbf{A}_3)^{\mathrm{T}}\mathbf{E}_1, (\mathbf{X}-\mathbf{A}_3)^{\mathrm{T}}\mathbf{E}_2)^{\mathrm{T}}\,, \tag{16}$$

of vectors $\mathbf{A}_1-\mathbf{A}_3$, $\mathbf{A}_2-\mathbf{A}_3$, $\mathbf{X}-\mathbf{A}_3$ on to the plane defined by points $\mathbf{A}_i$, solve the equation

$$\begin{pmatrix} U_1^1 & U_2^1 \\ U_1^2 & U_2^2 \end{pmatrix}\begin{pmatrix} w_1 \\ w_2 \end{pmatrix}=\begin{pmatrix} V^1 \\ V^2 \end{pmatrix}, \tag{17}$$

and calculate $w_3=1-w_1-w_2$.

The use of eq. (17) in computing local coordinates w_1, w_2, w_3 cannot prevent us from the inaccuracy caused by rounding errors. Let us consider two neighbouring ray cells and a point $\mathbf{X}$ situated on the side that is common to both ray cells. Note that the side is usually not planar, because it is defined by four points on rays. For such a point $\mathbf{X}$, one of the coordinates w_1, w_2, w_3 is roughly zero; let it be w_i in the first cell and w_j in the second. Theoretically, either both w_i and w_j are equal to zero, or w_i times w_j is less than zero. Due to the rounding errors this is not true for point $\mathbf{X}$ located very close to the side of the ray cell, because we have used coordinates of all the points forming the corresponding ray cell to compute local coordinates w_i and w_j, and not just the coordinates of the four points common to both the neighbouring cells.

Point $\mathbf{X}$ may thus numerically be located either in both the cells or in one of the cells or in no cell (see Fig. 7). On a PC, the relative width of the volume along the side of the ray cell, where the rounding errors described cause inaccuracy, is about 10^{-6} in order of magnitude as compared to the coordinates of the vertices of the side.

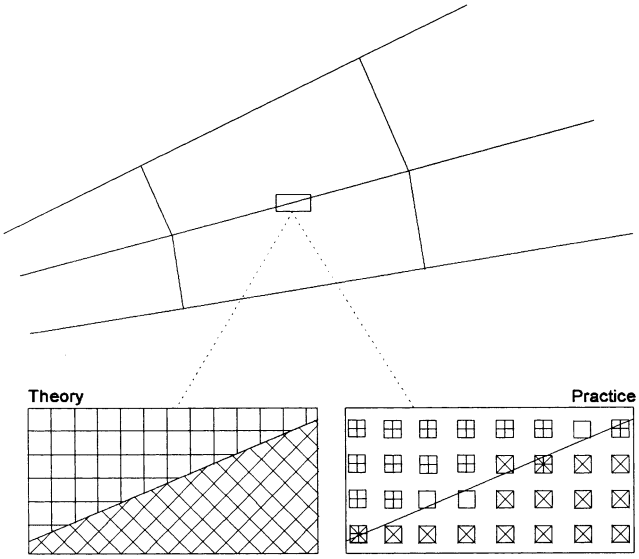

Figure 7. Theory and practice of the determination of which of the two ray cells a given point is located in. Theoretically, we should be able to decide correctly for any point whether it is located inside the ray cell under consideration. In practice, the rounding errors, for points located within a thin strip along the side separating the two ray cells, may make a decision based on eqs (6) and (17) incorrect. An additional check of these errors is necessary.

Assume that point $\mathbf{X}$ is very close to side $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{C}_1$, $\mathbf{C}_2$. We can get rid of the numerical problems described by constructing virtual ray cell $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{B}'_3$, $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}'_3$. First we construct points $\mathbf{S}=(\mathbf{B_1}+\mathbf{B_2})/2$ and $\mathbf{T}=(\mathbf{C_1}+\mathbf{C_2})/2$. We then get the coordinates of points $\mathbf{B}'_3=\mathbf{S}+(\mathbf{B_2}-\mathbf{B_1})\times(\mathbf{T}-\mathbf{S})$ and $\mathbf{C}'_3=\mathbf{T}+(\mathbf{C_2}-\mathbf{C_1})\times(\mathbf{T}-\mathbf{S})$, where $\times$ is the vector product. We compute the local coordinates of point $\mathbf{X}$ in the virtual ray cell. According to their value we can unambiguously decide about the position of point $\mathbf{X}$ with respect to the side $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{C}_1$, $\mathbf{C}_2$. Because the values of the local coordinates in the virtual ray cell are influenced only by the coordinates of the four points located on the given side, the rounding errors should be exactly the same in both the neighbouring real ray cells. We must only order points $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{C}_1$, $\mathbf{C}_2$ equally for both the cells when constructing the virtual cells.

4 INTERPOLATION WITHIN A RAY CELL

4.1 Bilinear interpolation

The simplest interpolation to fit the functional values at six points $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{B}_3$, $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$ is the bilinear interpolation with respect to w_B and (w_1, w_2). Linear interpolation

$$\tau^{\mathbf{A}_k}=w_B\tau^{\mathbf{B}_k}+w_C\tau^{\mathbf{C}_k}\,,\quad w_C=1-w_B\,, \tag{18}$$

with respect to w_B is used to obtain the values $\tau^{\mathbf{A}_k}$ of an interpolated quantity τ at points $\mathbf{A}_k$ from known values $\tau^{\mathbf{B}_k}$ and $\tau^{\mathbf{C}_k}$ at the vertices of the ray cell. Linear interpolation

$$\tau^{\mathbf{X}}=w_1\tau^{\mathbf{A}_1}+w_2\tau^{\mathbf{A}_2}+w_3\tau^{\mathbf{A}_3} \tag{19}$$

with respect to (w_1, w_2) is then used to obtain the values at point $\mathbf{X}$. This interpolation fits the functional values at the six vertices of the ray cell. Since the values at each cell face are dependent on the values at the vertices of the face only, the interpolated values are continuous across the cell faces.

This interpolation seems convenient in interpolating all the quantities resulting from ray tracing, except for the traveltime, which should be interpolated more accurately (see Fig. 8).

4.2 Bicubic interpolation

Assume that partial derivatives

$$p^i = \frac{\partial \tau}{\partial x^i} \tag{20}$$

are known in addition to the functional values of τ at six vertices $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{B}_3$, $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$ of the ray cell. We may then wish to use the bicubic interpolation with respect to w_B and (w_1, w_2) to fit both the functional values and the first partial derivatives at the six vertices. To fit the functional values, we define the cubic function

$$a(w) = w^2(3-2w) \tag{21}$$

with functional values $a(0)=0$, $a(1)=1$ and first derivatives $a'(0)=0$, $a'(1)=0$ at the endpoints of the interval $\langle 0, 1\rangle$. To fit the derivatives, we define the cubic function

$$b(w_B) = (w_B)^2 \sum_{i=1}^{3} p^i_{\mathbf{B}}[A^i(w_B) - B^i] \tag{22}$$

with functional values $b(0)=0$, $b(1)=0$ and derivatives $b'(0)=0$, $b'(1)=\Sigma_{i=1}^{3} p^i_{\mathbf{B}}[B^i - C^i]$ at the endpoints of the interval $\langle 0, 1\rangle$. Here

$$\mathbf{A}(w_B) = w_B\mathbf{B} + (1-w_B)\mathbf{C} \tag{23}$$

(see eq. 2). Analogously we define function

$$b(w_C) = (w_C)^2 \sum_{i=1}^{3} p^i_{\mathbf{C}}[A^i(w_C) - C^i]\,. \tag{24}$$

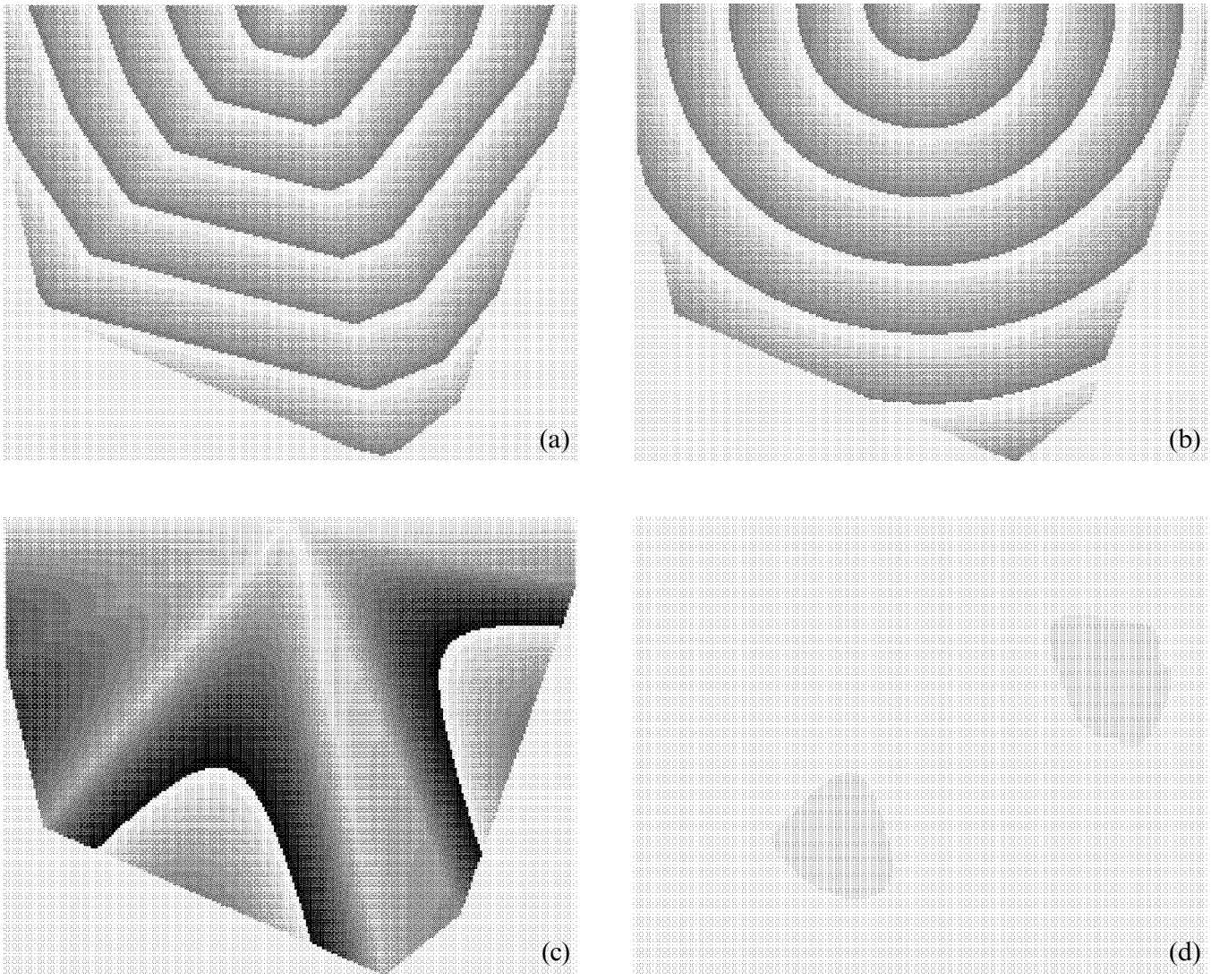

Figure 8. (a) Traveltimes in a homogeneous model interpolated bilinearly within wide 3-D ray tubes using eq. (19). One grey-scale circle corresponds to 0.2 s. (b) Traveltimes in a homogeneous model interpolated within the same ray tubes using bicubic interpolation (27). The maximum traveltime is about 2.7 s. (c) Differences of the bilinearly interpolated traveltimes from the exact solution. The maximum difference is about 0.308 s. (d) Differences of the bicubically interpolated traveltimes from the exact solution. The maximum difference is about 0.017 s.

Cubic interpolations

$$\tau^{\mathbf{A}_k} = a(w_B)\tau^{\mathbf{B}_k} + a(w_C)\tau^{\mathbf{C}_k} + (w_B)^2 \sum_{i=1}^{3} p^i_{\mathbf{B}_k}[A^i_k - B^i_k] + (w_C)^2 \sum_{i=1}^{3} p^i_{\mathbf{C}_k}[A^i_k - C^i_k] \quad (25)$$

and

$$p^i_{\mathbf{A}_k} = a(w_B)p^i_{\mathbf{B}_k} + a(w_C)p^i_{\mathbf{C}_k}, \quad i = 1, 2, 3, \quad (26)$$

along the edges of the ray cell, parametrized by w_B, are used to obtain functional values $\tau^{\mathbf{A}_k}$ at points $\mathbf{A}_k$ together with their first derivatives $p^i_{\mathbf{A}_k}$. The derivatives will be used for the interpolation within triangle $\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3$.

Cubic interpolation within triangle $\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3$,

$$\tau^{\mathbf{X}} = \sum_{k=1}^{3} a(w_k)\tau^{\mathbf{A}_k} + \sum_{k=1}^{3} (w_k)^2 \sum_{i=1}^{3} p^i_{\mathbf{A}_k}[X^i - A^i_k] + w_1w_2w_3\xi, \quad (27)$$

with respect to (w_1, w_2), is then used to obtain the values at point $\mathbf{X}$. The three functional values and six first partial derivatives at the vertices of planar triangle $\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3$ are fitted by the cubic function of w_1 and w_2. Since a cubic function of two variables has 10 coefficients, there is a free coefficient ξ in the last equation. Coefficient ξ corresponds to the basis function $w_1w_2w_3$, which has both the functional values and the first derivatives at the vertices $\mathbf{A}_k$ equal to zero.

Let us now check the interpolation of simple linear, quadratic and cubic functions. First, assume a simple linear function

$$\tau(w_1, w_2) = w_3, \quad (28)$$

where $w_3 = 1 - w_1 - w_2$. Then, as derived in the Appendix,

$$\tau^{\mathbf{X}} = w_3 + (\xi - 2)w_1w_2w_3. \quad (29)$$

Analogously for two other independent linear functions,

$$\tau(w_1, w_2) = w_1 \quad \text{and} \quad \tau(w_1, w_2) = w_2. \quad (30)$$

Now assume a simple quadratic function

$$\tau(w_1, w_2) = w_3(1 - w_3). \quad (31)$$

The interpolation then yields

$$\tau^{\mathbf{X}} = w_3(1 - w_3) + (\xi - 2)w_1w_2w_3 \quad (32)$$

(see the Appendix). Analogously for both of the other independent quadratic functions,

$$\tau(w_1, w_2) = w_1(1 - w_1) \quad \text{and} \quad \tau(w_1, w_2) = w_2(1 - w_2). \quad (33)$$

Coefficient ξ must depend linearly on the given interpolated quantities $\tau^{\mathbf{A}_k}$, $k = 1, 2, 3$, and $\Sigma_{i=1}^{3} p^i_{\mathbf{A}_k}[A^i_l - A^i_k]$, $k, l = 1, 2, 3$, $k \neq l$. The linear combination must be symmetric with respect to the permutation of points $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{A}_3$. There are only two linearly independent symmetric linear combinations of the given quantities,

$$\sum_{k=1}^{3} \tau^{\mathbf{A}_k} \quad \text{and} \quad \sum_{k=1}^{3}\sum_{l=1}^{3}\sum_{i=1}^{3} p^i_{\mathbf{A}_k}[A^i_l - A^i_k]. \quad (34)$$

There is no other linear combination of the given interpolated quantities, symmetric with respect to points $\mathbf{A}_1$, $\mathbf{A}_2$ and $\mathbf{A}_3$.

To interpolate linear and quadratic functions exactly, we thus have to take

$$\xi = 2\sum_{k=1}^{3} \tau^{\mathbf{A}_k} + \frac{1}{2}\sum_{k=1}^{3}\sum_{l=1}^{3}\sum_{i=1}^{3} p^i_{\mathbf{A}_k}[A^i_l - A^i_k]. \quad (35)$$

Here the first term is necessary to interpolate linear functions exactly, and the second term is added to interpolate quadratic functions exactly also.

Now assume a simple cubic function

$$\tau(w_1, w_2) = (w_2 - w_1)[(w_2 - w_1)^2 - 1]. \quad (36)$$

The interpolation then yields

$$\tau^{\mathbf{X}} = (w_2 - w_1)[(w_2 - w_1)^2 - 1] \quad (37)$$

(see Appendix), and analogously for the other two independent cubic functions,

$$\tau(w_1, w_2) = (w_3 - w_2)[(w_3 - w_2)^2 - 1] \quad \text{and} \quad \tau(w_1, w_2) = (w_1 - w_3)[(w_1 - w_3)^2 - 1]. \quad (38)$$

The last independent cubic function,

$$\tau(w_1, w_2) = w_1w_2w_3, \quad (39)$$

has zero functional values and first derivatives at all three vertices and thus yields

$$\tau^{\mathbf{X}} = 0. \quad (40)$$

There is no way of removing this inaccuracy in the interpolation of cubic functions without additional information.

The interpolation fits the values and the first spatial partial derivatives at the six vertices of the ray cell. The functional values at each cell face depend on the functional values and derivatives at the vertices of the face only. The interpolated values are thus continuous across the cell faces. The gradient of the interpolated function is continuous at the triangular cell faces and along the cell edges, but may be discontinuous at the tetragonal cell faces separating ray tubes. The interpolation fits linear and quadratic functions exactly.

5 NUMERICAL EXAMPLE

The interpolation method described was coded in Fortran 77 and was incorporated into the ray tracing program package CRT. A computation in the model called '98' is presented here as an example of the interpolation of ray theory traveltimes on a dense rectangular grid. The sample synthetic model '98' was created during the technical session of the consortium 'Seismic Waves in Complex 3-D Structures' meeting in June 1998. The aim of the work done was not to create a realistic model, but to

demonstrate the capabilities of program packages MODEL for the specification of heterogeneous seismic models and CRT for complete ray tracing in the models.

Model '98' is composed of three layers separated by two interfaces. The upper interface is curved; the lower one is formed by a dipping plane. The inclusion in the middle layer is approximately spherical. The first layer represents water with a constant *P*-wave velocity and zero *S*-wave velocity. In the other model blocks, the *P*-wave velocity has a vertical gradient and smaller lateral gradients and the *S*-wave velocity is a linear function of the *P*-wave velocity (see Fig. 9).

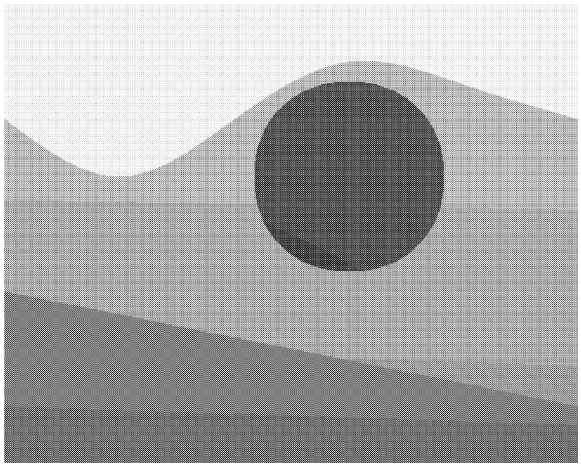

Figure 9. *P*-wave velocity cross-section through the middle of model '98'. The *P*-wave velocity is lowest at the top water layer and highest within the spherical body.

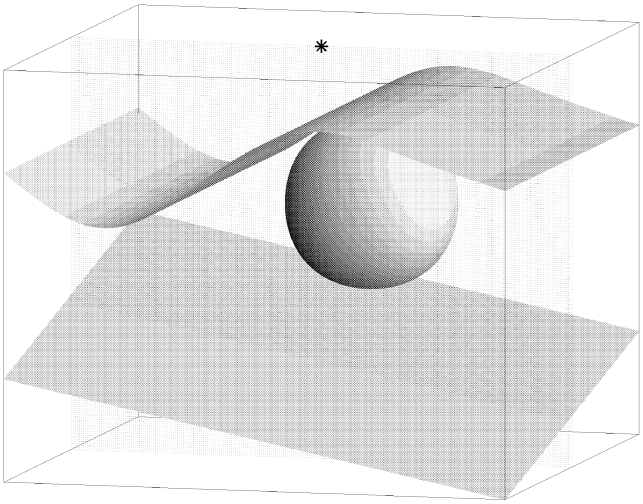

Figure 10. Structural interfaces, point source and 2-D receiver grid in model '98'.

The ray theory traveltimes are computed for a vertical 2-D grid of 251×201 points embedded in the 3-D model. The point source is located on the same plane as the receivers at a depth of 0.001 km (see Fig. 10). The elementary wave refracted at each interface is considered. Only structural interfaces affect ray histories.

17604 rays are computed, including the rays used for interpolation and auxiliary rays. The computation of rays takes 6 min on a Pentium 200 MHz PC running under MS DOS; the interpolation takes 5 min on the same computer.

The computed multivalued traveltime maps may be sorted according to the arrival time (that is, the first arrival at the gridpoints, the second arrival at the gridpoints, etc.) (see Fig. 11). Another possibility is to convert the computed multivalued traveltime maps into single-valued maps sorted according to ray histories (see Fig. 12). The single-valued traveltime maps sorted according to ray histories may be merged together manually. One result that may be obtained is shown in Fig. 13. Note that some traveltimes were omitted during the merging process, so there are only three resulting pictures.

6 CONCLUSIONS

We have proposed a method for the interpolation of ray theory traveltimes and other quantities to gridpoints of dense rectangular grids. Ray tubes are decomposed into individual ray cells in the process, which ensures that the sides of ray cells in neighbouring ray tubes coincide, and no receiver points are omitted along these sides. The decomposition also prevents interpolation across structural interfaces.

Using the scheme, all the quantities computed along rays may be interpolated. The scheme requires the values of the interpolated quantities only at the vertices of the corresponding ray cell. It incorporates both the decision whether a receiver lies in the ray cell and the interpolation to the receiver. The scheme is applicable to all ray cells formed by six, five or four points. Further investigation should concentrate on numerical tests of the scheme and on comparisons with other schemes.

The numerical example presented shows the applicability of the method to a 3-D model with interfaces.

ACKNOWLEDGMENTS

This research has been supported by the consortium 'Seismic Waves in Complex 3-D Structures'.

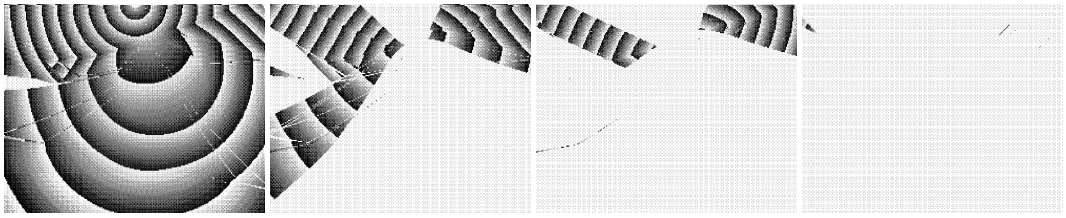

Figure 11. The single-valued traveltime maps sorted according to the arrival time.

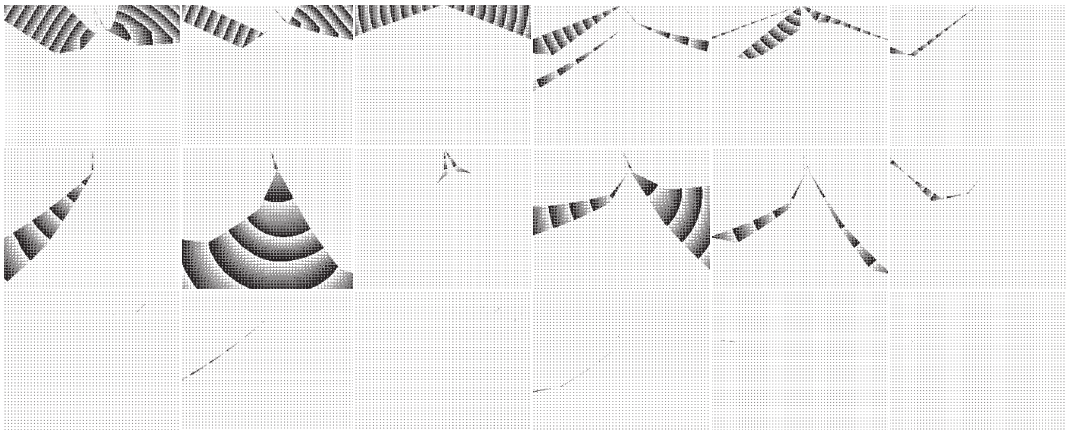

Figure 12. The single-valued traveltime maps sorted according to ray histories.

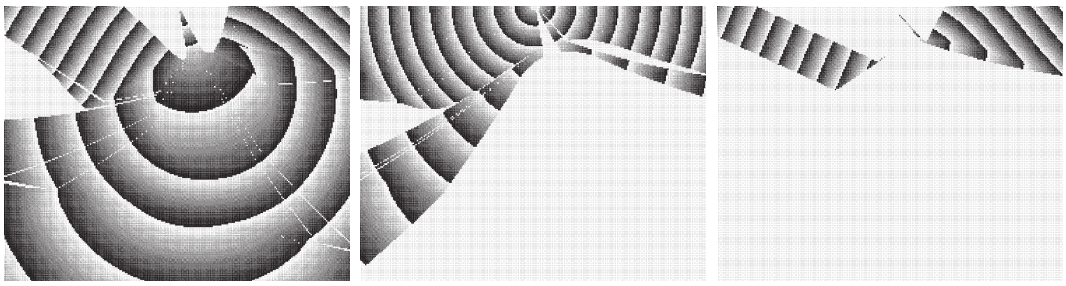

Figure 13. The single-valued traveltime maps sorted according to the ray histories from Fig. 12, merged into the three sections. Some computed traveltimes were omitted.

REFERENCES

Bulant, P., 1996. Two-point ray tracing in 3-D, *Pure appl. Geophys.,* **148,** 421–446.

Bulant, P., 1997a. Controlled initial-value ray tracing and two-point ray tracing in 3D, *Extended Abstracts 59th EAGE Conference, Geneva,* P025, Eur. Assoc. Geoscientists & Engineers, Zeist.

Bulant, P., 1997b. Two-point ray tracing and controlled initial-value ray tracing in 3-D heterogeneous block structures, *Expanded Abstracts 67th Annual Meeting, Dallas,* pp. 1727–1730, SEG, Tulsa.

Bulant, P., 1999. Prismatic ray cells versus tetrahedra—numerical tests of interpolation methods, in *Seismic Waves in Complex 3-D Structures,* Report 8, pp. 69–70, Dept Geophysics, Charles University, Prague.

Lucio, P.S., Lambaré, G. & Hanyga, A., 1996. 3D multivalued travel time and amplitude maps, *Pure appl. Geophys.,* **148,** 449–479.

Moser, T.J. & Pajchel, J., 1997. Recursive seismic ray modelling: applications in inversion and VSP, *Geophys. Prospect.,* **45,** 885–908.

Press, W.H., Teukolsky, S.A., Vetterling, W.T. & Flannery, B.P. *Numerical Recipes in Fortran 77: The Art of Scientific Computing,* 2nd edn, Cambridge University Press, Cambridge.

Vinje, V., Iversen, E., Gjøystdal, H. & Åstebøl, K., 1996a. Estimation of multivalued arrivals in 3D models using wave front construction—Part I, *Geophys. Prospect.,* **44,** 819–842.

Vinje, V., Iversen, E., Gjøystdal, H. & Åstebøl, K., 1996b. Part II: tracing and interpolation, *Geophys. Prospect.,* **44,** 843–858.

APPENDIX A: DERIVATION OF EQS (29), (32) AND (37)

For a given function $\tau(w_1, w_2)$, we introduce complementary parametrizations

$$\begin{aligned}
\tau_{12}(w_1, w_2) &= \tau(w_1, w_2)\,,\\
\tau_{13}(w_1, w_3) &= \tau(w_1, 1-w_1-w_3)\,,\\
\tau_{23}(w_2, w_3) &= \tau(1-w_2-w_3, w_2)\,.
\end{aligned} \tag{A1}$$

Then

$$\begin{aligned}
p^i_{\mathbf{A}_1}[X^i - A^i_1] &= p^i_{\mathbf{A}_1} \sum_j w_j [A^i_j - A^i_1]\\
&= p^i_{\mathbf{A}_1} w_2 [A^i_2 - A^i_1] + p^i_{\mathbf{A}_1} w_3 [A^i_3 - A^i_1]\\
&= w_2 \frac{\partial \tau_{23}}{\partial w_2}(1, 0) + w_3 \frac{\partial \tau_{23}}{\partial w_3}(0, 1)
\end{aligned} \tag{A2}$$

and vice versa.

A1 Eq. (29)

Linear function

$$\tau(w_1, w_2) = w_3 \tag{28}$$

then has

$$\begin{aligned}
&\tau^{\mathbf{A}_1} = 0\,, \quad \tau^{\mathbf{A}_2} = 0\,, \quad \tau^{\mathbf{A}_3} = 1\,,\\
&p^i_{\mathbf{A}_1}[\mathbf{X}^i - \mathbf{A}^i_1] = w_3\,, \quad p^i_{\mathbf{A}_2}[\mathbf{X}^i - \mathbf{A}^i_2] = w_3\,, \quad p^i_{\mathbf{A}_3}[\mathbf{X}^i - \mathbf{A}^i_3] = w_3 - 1\,,
\end{aligned} \tag{A3}$$

and interpolation (27) yields

$$\begin{aligned}
\tau^{\mathbf{X}} &= (w_3)^2(3-2w_3) + (w_1)^2 w_3 + (w_2)^2 w_3 + (w_3)^2(w_3-1)\\
&\quad + w_1 w_2 w_3 \xi\\
&= w_3[w_3(2-w_3) + (w_1)^2 + (w_2)^2] + w_1 w_2 w_3 \xi\\
&= w_3[(1-w_1-w_2)(1+w_1+w_2) + (w_1)^2 + (w_2)^2]\\
&\quad + w_1 w_2 w_3 \xi\\
&= w_3 + (\xi - 2) w_1 w_2 w_3\,.
\end{aligned} \tag{A4}$$

A2 Eq. (32)

Quadratic function

$$\tau(w_1, w_2) = w_3(1 - w_3) \tag{31}$$

then has

$$\begin{aligned}
&\tau^{\mathbf{A}_1} = 0\,, \quad \tau^{\mathbf{A}_2} = 0\,, \quad \tau^{\mathbf{A}_3} = 0\,,\\
&p^i_{\mathbf{A}_1}[\mathbf{X}^i - \mathbf{A}^i_1] = w_3\,, \quad p^i_{\mathbf{A}_2}[\mathbf{X}^i - \mathbf{A}^i_2] = w_3\,, \quad p^i_{\mathbf{A}_3}[\mathbf{X}^i - \mathbf{A}^i_3] = 1 - w_3\,,
\end{aligned} \tag{A5}$$

and interpolation (27) yields

$$\begin{aligned}
\tau^{\mathbf{X}} &= (w_1)^2 w_3 + (w_2)^2 w_3 + (w_3)^2(1-w_3) + w_1 w_2 w_3 \xi\\
&= w_3[w_3(1-w_3) + (w_1)^2 + (w_2)^2] + w_1 w_2 w_3 \xi\\
&= w_3[(1-w_1-w_2)(w_1+w_2) + (w_1)^2 + (w_2)^2] + w_1 w_2 w_3 \xi\\
&= w_3(1-w_3) + (\xi - 2) w_1 w_2 w_3\,.
\end{aligned} \tag{A6}$$

A3 Eq. (37)

Cubic function

$$\tau(w_1, w_2) = (w_2 - w_1)[(w_2 - w_1)^2 - 1] \tag{36}$$

then has

$$\begin{aligned}
&\tau^{\mathbf{A}_1} = 0\,, \quad \tau^{\mathbf{A}_2} = 0\,, \quad \tau^{\mathbf{A}_3} = 0\,,\\
&p^i_{\mathbf{A}_1}[\mathbf{X}^i - \mathbf{A}^i_1] = 2(w_2 - w_1 + 1)\,,\\
&p^i_{\mathbf{A}_2}[\mathbf{X}^i - \mathbf{A}^i_2] = 2(w_2 - w_1 - 1)\,,\\
&p^i_{\mathbf{A}_3}[\mathbf{X}^i - \mathbf{A}^i_3] = -(w_2 - w_1)\,, \quad \xi = 0\,,
\end{aligned} \tag{A7}$$

and interpolation (27) yields

$$\begin{aligned}
\tau^{\mathbf{X}} &= 2(w_1)^2(w_2 - w_1 + 1) + 2(w_2)^2(w_2 - w_1 - 1)\\
&\quad - (w_3)^2(w_2 - w_1)\\
&= (w_2 - w_1)[2(w_1)^2 + 2(w_2)^2 - 2w_1 - 2w_2\\
&\quad - (1 - w_1 - w_2)^2]\\
&= (w_2 - w_1)[(w_1)^2 + (w_2)^2 - 2w_1 w_2 - 1]\\
&= (w_2 - w_1)[(w_2 - w_1)^2 - 1].
\end{aligned} \tag{A8}$$