Grid travel–time tracing: second–order method for the first arrivals in smooth media *

Luděk Klimeš

Department of Geophysics, Charles University, Ke Karlovu 3, 121 16 Praha 2, Czech Republic, Fax: +42-2-299272, E-mail: psencik@earn.cvut.cz

Summary

New computational scheme for calculating the first–arrival travel times on a rectangular grid of points is proposed. Whereas the former "finite–difference" eikonal solvers are the methods of the first order, the new proposed method is of the second–order accuracy. It means that the relative propagation–velocity error of calculated travel time is proportional to the second power of the grid step. That is why the method should be sufficiently accurate for all applications in smooth seismic models. On the other hand, the method is not, in its present form, designed for models with structural interfaces which make the method unstable and generate travel–time errors of the first order. The equations for appropriate evaluation of the errors of calculated travel times in order to check their accuracy are also presented, and the proposed method is compared with other two numerical methods. The method is developed, described, and demonstrated in 2-D, but may also be extended to general 3-D models with structural interfaces.

1 Introduction

In recent years, following the extension of random–access computer memory (RAM), many authors proposed the methods of calculating travel times on regular rectangular grids of points. The grids represent the discretized forms of seismic models, like in full–wave finite difference methods. As a rule, the travel–time calculating methods have been proposed for the first–arrival travel times, and may be divided into two classes:

(a) Methods interpolating already calculated travel times between gridpoints in order to evaluate the travel time in the closest next gridpoint. These methods are often called "finite–difference" eikonal–equation solvers, but the author would prefer some other common name, for instance *grid travel–time tracing* proposed here, because the ordinary meaning of "finite–differences" would have to be generalized considerably to include also these methods. These methods developed from the non–causal (and thus practically unusable) idea of Vidale (1988) to the stable method of the first order by Podvin & Lecomte (1991). The accuracy of these methods may considerably be increased by replacing the plane wavefront approximation by a method of higher order, if the updating scheme is made stable for the interpolation method.

(b) Methods based on shortest–path network algorithms. They developed from the less accurate methods of the orders between 0 and ⅔ by Nakanishi & Yamaguchi (1986), Moser (1989, 1991), and Saito (1989, 1990), to the self–adaptable algorithm of the first order by Klimeš & Kvasnička (1994).

The method proposed here follows grid travel–time tracing approach (a), and is based on the idea of calculating the *slowness vectors* at gridpoints together with *travel times*. In the proposed method, the travel times between two adjacent gridpoints are interpolated by an accurate second–order or third–order method. The point on the gridline segment, from which the ray propagates to the next gridpoint, is approximated with the

* Revised version. To be submitted to J. seism. Explor.

accuracy proportional to the squared grid step. The linear interpolation is used for the slowness. From the point on the gridline segment, the slowness vectors and travel times are approximated by expansions accurate up to terms with second partial travel–time derivatives. Thus, in a smooth media, the resulting travel–time error corresponds to the neglected second partial slowness derivatives, and is of the order of the squared grid step. Then the rays are approximated very accurately in smooth parts of a model, i.e. in the regions where the second partial slowness derivatives are kept within some reasonable finite limits. This is not the case of the first and second order interfaces, where the method has to be generalized by means of matching travel–times at both sides of the interface.

2 Algorithm

Slowness vectors are calculated at gridpoints together with *travel times*. It means that, in 2-D, instead of each slowness describing the discretized smooth model, 2 components of the slowness vector have to be stored in the memory. The length of the slowness vector is, of course, equal to the slowness and is not changed during the calculation. Before calculation, input velocities are transformed to slowness vectors of arbitrary orientations. During the updating process, the orientations of slowness vectors are step–by–step being corrected, without changing their lengths.

Thus, in 2-D smooth models, 4 storage locations are required per a gridpoint: 1 travel time, 2 components of the slowness vector, and 1 index in the queue. The queue contains the indices of the gridpoints to be updated or to initiate updating of other gridpoints.

The updating consists in finding the minimum travel time corresponding to the considered kinds of travel–time propagation. Under travel–time propagation we understand here the considered ray paths leading to the gridpoint, at which travel time is calculated, from the gridpoints or gridline segments where the travel time has already been determined.

Two different kinds of travel–time propagation are considered:
(**a**) the travel time field interpolated along the gridline segment between two adjacent gridpoints and propagating along rays perpendicular to the local wavefronts.
(**b**) diffractions at gridpoints.

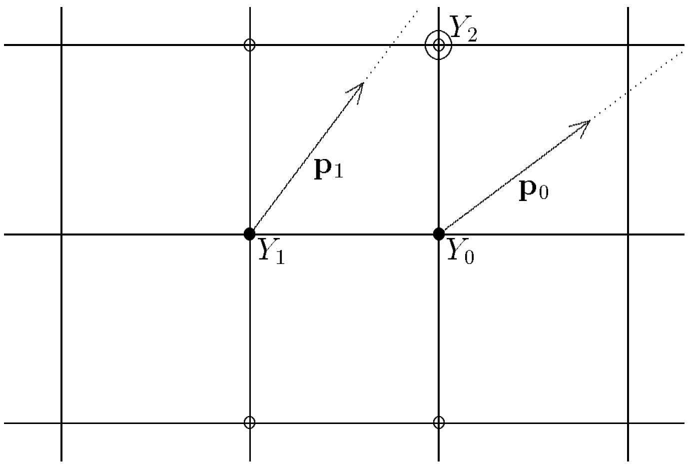

Figure 1. •'s denote two end gridpoints of the gridline segment along which the travel time has already been determined and which initiate updating; o's denote the end gridpoints of two nearest parallel gridline segments; ◯ denotes the gridpoint to be updated if the new travel time is less than the previously determined value.

Travel time fields of kind (**a**) are taken into account just in the region between two half–lines given by the slowness vectors. Only the end gridpoints of two nearest parallel gridline segments are allowed to be updated. See Figure 1.

Diffracted travel time fields (**b**) are taken into account at all 8 nearest gridpoints.

Source. For the sake of conciseness, we consider only a point source in this paper.

Initialization. When starting the algorithm, gridpoints surrounding the source are updated and put into the beginning of the queue. Under the surrounding gridpoints we understand here the intersections of gridlines passing up to the distance of $h + d$ from the point source. Here d is a small positive length not exceeding grid interval h in order. For example, the author presently applies $d = \frac{1}{2}h$. Updating consists in determination of travel times and slowness vectors, and is, of course, of type (**b**) .

It can be seen in Figure 1 that the travel–time increment at the updated gridpoint Y_2, with respect to the nearest gridpoint Y_0, should not be less than

$$d_{\min} = (\sqrt{2} - 1)\, h\, u_{\min} \quad , \tag{1}$$

where h is the grid interval and $u_{\min}$ is the minimum slowness in the model. Minimum increment (1) is reached for the point source situated very close to gridpoint Y_1, because gridpoint Y_2, in order to be updated, has to be situated inside the region between two half–lines given by the slowness vectors at Y_0 and Y_1. Moreover, if point Y_2 is so distant from the point source that it is not updated during initialization, the travel–time increment should not be less than

$$d_0 = \left[\sqrt{2}\,(h + d) - \sqrt{(h + d)^2 + d^2}\right] u_{\min} \quad . \tag{2}$$

The travel–time difference between points Y_1 and Y_2 could, in principle, be zero for the point source close to Y_0. However, if point Y_1 is surrounding the point source, it is updated during the initialization and located in the queue in front of Y_2 and is thus to initiate updating sooner. On the other hand, if Y_1 is not surrounding a point source, the travel–time difference between points Y_1 and Y_2 should be at least

$$d_1 = \left[2h + d - \sqrt{(h + d)^2 + h^2}\right] u_{\min} \quad , \tag{3}$$

which is greater than (2). Because the difference between (1) and (2) should not considerably influence the numerical efficiency of the algorithm, and because the above considerations are, to some extent, approximate, we suggest to use (1) as the lower bound of (2), independent of d.

Thus we may separate the gridpoints, according to first–arrival travel times, into intervals of length not exceeding $d_{\min}$. Then the node being updated belongs to a higher interval than the nodes Y_0 and Y_1 initiating updating, and the `INTERVAL` algorithm by Klimeš & Kvasnička (1994) may be used to select the gridpoints to initiate updating of the neighbouring gridpoints.

The `INTERVAL` *algorithm.* In the beginning, only the gridpoints with travel time determined during the initialization are stored in an unsorted queue. The queue may be represented, e.g., by an array Q in which the indices of the gridpoints are being stored. Once the gridpoint has been updated, it is added to the queue. First, updating is initiated by the nodes in the queue belonging to the lowest travel–time interval of size $d_{\min}$. The gridpoints which already have initiated updating are removed from the queue. When all gridpoints of the interval are removed from the queue, we progress to new travel–time interval of length $d_{\min}$ and the procedure continues until the queue is empty.

Now the reader have noticed that we spoke of a single gridpoint initiating updating whereas the travel–time propagation (**a**) is initiated by a couple of neighbouring gridpoints. If a gridpoint is selected by the `INTERVAL` algorithm to initiate updating, all 4 neighbouring nodes (in 2-D) are being searched for travel time not exceeding travel time at the selected gridpoint. The updating of kind (**a**) is then started from all gridline segments Y_0Y_1 connecting the gridpoint initiating updating with a gridpoint of less or equal travel time. There should be, in most cases but not always, 2 such gridlines per each gridpoint in 2-D.

Each gridline Y_0Y_1 may initiate updating at most at 2 of 4 end gridpoints of the nearest parallel gridline segments. There will be, in most cases but not always, 0 or 1 such point Y_2 per each gridline segment Y_0Y_1 in 2-D, one half in average. Two points Y_2 per gridline segment Y_0Y_1 are expected not to be frequent in smooth models.

3 Travel–time calculation

3.1 Propagation of type (a)

Let us assume that Y_0 is the endpoint of gridline segment Y_0Y_1 initiating updating, nearest to the updated point Y_2, whereas Y_1 is the more distant one. For the travel–time interpolation, a local coordinate system may be introduced by the following simple transformation of a general 2-D Cartesian coordinate system connected with the grid: the origin is moved to point Y_0, and, if the gridline segment Y_0Y_1 is not parallel with the first coordinate axis, the two coordinates are exchanged. Then endpoints Y_0 and Y_1 of the gridline segment initiating updating, and gridpoint Y_2 to be updated have coordinates

$$\mathbf{x}_0 = (0,0) \ , \quad \mathbf{x}_1 = (\Delta x_1, 0) = (\pm h, 0) \ , \quad \mathbf{x}_2 = (0, \Delta x_2) = (0, \pm h) \ , \tag{4}$$

respectively. We also denote by

$$\tau_0 \ , \ \mathbf{p}_0 = (p_{10}, p_{20}) \ , \quad \tau_1 \ , \ \mathbf{p}_1 = (p_{11}, p_{21}) \ , \quad \tau_2 \ , \ \mathbf{p}_2 = (p_{12}, p_{22}) \tag{5}$$

the corresponding travel times and slowness vectors in the local coordinate system.

3.1.1 Linear slowness interpolation

Slowness is linearly interpolated inside triangle $Y_0Y_1Y_2$, from slowness u_0 at Y_0, slowness u_1 at Y_1, and u_2 at Y_2,

$$u(x_1, x_2) = u_0 + U_1x_1 + U_2x_2 \quad , \tag{6}$$

with

$$\Big(U_1, U_2\Big) = \Big(\frac{u_1 - u_0}{\Delta x_1}, \frac{u_2 - u_0}{\Delta x_2}\Big) \tag{7}$$

being the slowness gradient.

3.1.2 Backprojection of the updated gridpoint

First, the backprojection $\tilde{Y}_2$ of the updated gridpoint Y_2 onto the gridline segment Y_0Y_1 initiating updating is determined, see Figure 2. The point of intersection of two straight lines given by the slowness vectors is selected as centre C of the central projection. It means that the back–projected point lies on the straight line connecting centre C and gridpoint Y_2 to be updated.

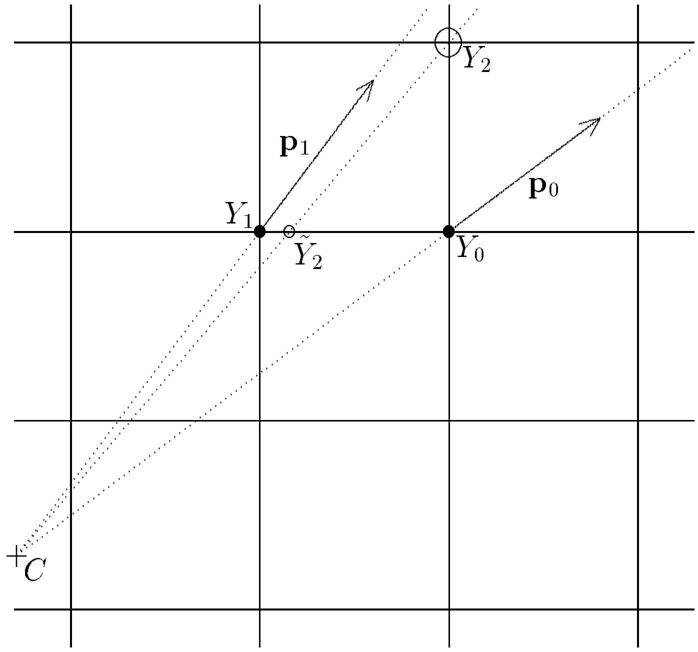

Figure 2. Y_0 and Y_1 are two end gridpoints of the gridline segment along which the travel time has already been determined and which initiate updating; Y_2 is the gridpoint to be updated if the new travel time is less than the previously determined value; C is the point of intersection of straight lines given by the slowness vectors; $\tilde{Y}_2$ is the backprojection of the gridpoint to be updated, i.e. the point at which travel time is interpolated.

3.1.3 Cubic or quadratic travel–time interpolation

Gridpoint Y_2 cannot be updated unless

$$\frac{p_{20}}{\Delta x_2} > 0 \quad \text{and} \quad \frac{p_{21}}{\Delta x_2} > 0 \quad . \tag{8}$$

If inequalities (8) are in force, the projections of points Y_0 and Y_1 onto the parallel gridline through Y_2 are

$$\tilde{\mathbf{x}}_0 = \Big(\tilde{x}_{10}, \Delta x_2\Big) = \Big(\frac{p_{10}\Delta x_2}{p_{20}}, \Delta x_2\Big) \ , \quad \tilde{\mathbf{x}}_1 = \Big(\tilde{x}_{11}, \Delta x_2\Big) = \Big(\Delta x_1 + \frac{p_{11}\Delta x_2}{p_{21}}, \Delta x_2\Big) \ . \tag{9}$$

Point Y_2 can be updated if it is situated in between the projections,

$$\frac{\tilde{x}_{10}}{\Delta x_1} < 0 \quad \text{and} \quad 0 < \frac{\tilde{x}_{11}}{\Delta x_1} \quad . \tag{10}$$

The backprojection $\tilde{Y}_2$ of point Y_2 is then

$$\tilde{\mathbf{x}}_2 = \Big(\tilde{x}_{12}, 0\Big) = \Big(\frac{\Delta x_1 \tilde{x}_{10}}{\tilde{x}_{10} - \tilde{x}_{11}}, 0\Big) \quad . \tag{11}$$

3.1.3.1 Cubic travel–time interpolation

Travel time between points Y_0 and Y_1 may be interpolated by the third–order Hermite polynomial interpolation with respect to $\tilde{x}_{12}$,

$$\tau(\tilde{x}_{12}) = (2a_0 + 1)a_1 a_1(\tau_0 - \tau_1) + \tau_1 + a_0 a_1(a_1 p_{10} + a_0 p_{11})\Delta x_1 \quad , \tag{12}$$

where

$$a_0(\tilde{x}_{12}) = \frac{\tilde{x}_{12}}{\Delta x_1} \ , \quad a_1(\tilde{x}_{12}) = a_0(\tilde{x}_{12}) - 1 \ . \tag{13}$$

If the backprojection coincides with Y_0, $a_0 = 0$, and if the backprojection coincides with Y_1, $a_1 = 0$. Taking into account that the derivatives of both a_0 and a_1 with respect to $\tilde{x}_{12}$ equal $(\Delta x_1)^{-1}$, the first and second derivatives of (12) are

$$p_1(\tilde{x}_{12}) = 6a_0 a_1(\tau_0 - \tau_1)(\Delta x_1)^{-1} + (2a_0 + a_1)a_1 p_{10} + (a_0 + 2a_1)a_0 p_{11} \tag{14}$$

and

$$T_{11}(\tilde{x}_{12}) = 6(a_0 + a_1)(\tau_0 - \tau_1)(\Delta x_1)^{-2} + 2[(a_0 + 2a_1)p_{10} + (2a_0 + a_1)p_{11}](\Delta x_1)^{-1} \ . \tag{15}$$

If there is an error δp_1 in p_{10} and p_{11}, this error is projected through equation (14) onto the new slowness–vector components p_1, with a multiplication factor of

$$q = (2a_0 + a_1)a_1 + (a_0 + 2a_1)a_0 = 1 + 6a_0 a_1 \quad . \tag{16}$$

For multiplication factors q close to 1, error δp_1 in the slowness–vector is accumulated along rays, and the accumulated error is projected by means of equation (12) onto the travel–time, see Section 4.1.

3.1.3.2 Quadratic travel–time approximation

The accumulation of error δp_1 in the slowness–vector may be avoided by a modification of interpolation (12). If average slowness–vector component $\frac{1}{2}(p_{10} + p_{11})$ is replaced in (12) by the difference $(\tau_1 - \tau_0)(\Delta x_1)^{-1}$, while considering the wavefront curvature described by difference $p_{11} - p_{10}$ of slowness–vector components, equation (12) is turned into quadratic approximation

$$\tau(\tilde{x}_{12}) = \tau_0 + a_0(\tau_1 - \tau_0) + \tfrac{1}{2}a_0 a_1(p_{11} - p_{10})\Delta x_1 \quad . \tag{17}$$

Taking into account that the derivatives of both a_0 and a_1 with respect to $\tilde{x}_{12}$ equal $(\Delta x_1)^{-1}$, the first and second derivatives of (17) are

$$p_1(\tilde{x}_{12}) = (\tau_1 - \tau_0)(\Delta x_1)^{-1} + \tfrac{1}{2}(a_0 + a_1)(p_{11} - p_{10}) \tag{18}$$

and

$$T_{11}(\tilde{x}_{12}) = (p_{11} - p_{10})(\Delta x_1)^{-1} \ . \tag{19}$$

Although quadratic approximation (17) causes local relative travel–time error of the order of h^2 not contained in (12), it is globally usually more stable and accurate than (12) because of avoiding accumulation of the error in orientation of the slowness vector. For example, compare Figures 3b and 3c of Section 6.

3.1.4 Spherical correction

The above cubic interpolation or quadratic approximation is sufficiently accurate if the distance of the centre C of the central projection,

$$x_2^C = \frac{\Delta x_1}{\left(\frac{p_{10}}{p_{20}} - \frac{p_{11}}{p_{21}}\right)} \quad , \quad x_1^C = \frac{p_{10}}{p_{20}}\, x_2^C \quad , \tag{20}$$

is sufficiently distant from gridline Y_0Y_1. In practice, "sufficiently distant" may mean at least N grid intervals, with given N,

$$N \left| \frac{p_{10}}{p_{20}} - \frac{p_{11}}{p_{21}} \right| < 1 \quad . \tag{21}$$

The value of N influences, for instance, the accuracy in the vicinity of the point source. Although the cubic interpolation is very accurate for the point source at a gridpoint even without this restriction, for general source position and possibly in some other situations, the value of N should not be less than 2. On the other hand, the demands of N greater than 20 are unlike even for very accurate testing calculations. As we shall see below, great N slows down the computation, but not considerably. That is why N need not be chosen too small. The author recommends the values of N from 10 to 20, believing that $N = 10$ should be sufficient for all contemporary seismological applications.

If inequality (21) is not in force, cubic interpolation or quadratic approximation of spherical wavefront may not be sufficiently accurate. In such a case, a reference travel–time distribution along gridline segment Y_0Y_1, resembling spherical wavefronts, may be introduced in order to interpolate only the travel–time residuals with respect to the reference solution.

In accordance with the central projection employed above, see Figure 2, a simple reference travel–time distribution is introduced: slowness vectors along gridline segment Y_0Y_1 are assumed to have the directions of the projecting straight lines passing through centre C of the central projection. Since the length of the slowness vector equals slowness

$$u(x_1, 0) = u_0 + U_1 x_1 \quad , \tag{22}$$

linearly interpolated along the gridline, the assumption yields the travel–time derivative along the gridline,

$$\mathring{p}_1(x_1) = \frac{x_1 - x_1^C}{r(x_1)}\, u(x_1, 0) \quad , \tag{23}$$

where

$$r(x_1) = \sqrt{(x_1 - x_1^C)^2 + (x_2^C)^2} \tag{24}$$

is the distance from centre C. Reference travel time is then the integral of its derivative (23),

$$\mathring{\tau}(x_1) = \tfrac{1}{2}\left\{ r(x_1)\left[u(x_1,0) + u(x_1^C,0)\right] - U_1 \left[x_2^C\right]^2 \ln\left(r(x_1) + x_1 - x_1^C\right)\right\} \quad . \tag{25}$$

The second reference travel–time derivative

$$\mathring{T}_{11}(x_1) = \frac{[x_2^C]^2}{[r(x_1)]^3}\, u(x_1,0) + \frac{x_1 - x_1^C}{r(x_1)}\, U_1 \tag{26}$$

is the result of differentiating (23).

The travel–time and first slowness–vector component residuals are

$$\tilde{\tau}_0 = \tau_0 - \mathring{\tau}(0)\ ,\ \tilde{p}_{10} = 0\ ,\quad \tilde{\tau}_1 = \tau_1 - \mathring{\tau}(\Delta x_1)\ ,\ \tilde{p}_{11} = 0\ . \tag{27}$$

3.1.4.1 Spherical correction with cubic interpolation

If choosing cubic interpolation of travel times described in Section 3.1.3.1, the travel–time residuals (27) may be interpolated by cubic interpolation (12),

$$\tilde{\tau}(\tilde{x}_{12}) = (2a_0+1)a_1a_1(\tilde{\tau}_0 - \tilde{\tau}_1) + \tilde{\tau}_1 \quad , \tag{28}$$

$$\tilde{p}_1(\tilde{x}_{12}) = 6a_0a_1(\tilde{\tau}_0 - \tilde{\tau}_1)(\Delta x_1)^{-1} \quad , \tag{29}$$

$$\tilde{T}_{11}(\tilde{x}_{12}) = 6(a_0+a_1)(\tilde{\tau}_0 - \tilde{\tau}_1)(\Delta x_1)^{-2} \quad . \tag{30}$$

3.1.4.2 Spherical correction with linear approximation

If choosing quadratic approximation of travel times described in Section 3.1.3.2, the travel–time residuals (27) should be approximated by linear interpolation

$$\tilde{\tau}(\tilde{x}_{12}) = a_1(\tilde{\tau}_0 - \tilde{\tau}_1) + \tilde{\tau}_1 \quad , \tag{31}$$

$$\tilde{p}_1(\tilde{x}_{12}) = (\tilde{\tau}_0 - \tilde{\tau}_1)(\Delta x_1)^{-1} \quad , \tag{32}$$

$$\tilde{T}_{11}(\tilde{x}_{12}) = 0 \quad . \tag{33}$$

The resulting interpolated quantities at the backprojection $\tilde{Y}_2$ of updated gridpoint Y_2 are

$$\tau(\tilde{x}_{12}) = \mathring{\tau}(\tilde{x}_{12}) + \tilde{\tau}(\tilde{x}_{12}) \quad , \tag{34}$$

$$p_1(\tilde{x}_{12}) = \mathring{p}_1(\tilde{x}_{12}) + \tilde{p}_1(\tilde{x}_{12}) \quad , \tag{35}$$

and

$$T_{11}(\tilde{x}_{12}) = \mathring{T}_{11}(\tilde{x}_{12}) + \tilde{T}_{11}(\tilde{x}_{12}) \quad . \tag{36}$$

Residual quantities $\tilde{\tau}$, $\tilde{p}_1$, and $\tilde{T}_{11}$ in (34)–(36) are given by (28)–(30) or (31)–(33), respectively.

3.1.5 Slowness–vector extrapolation

We denote $\tilde{u}_2 = u(\tilde{x}_{12}, 0)$ the slowness at the backprojection $\tilde{Y}_2$. If

$$(\tilde{u}_2)^2 > (p_1)^2 \quad , \tag{37}$$

that should be in force for all reasonably discretized models, the second slowness component at $\tilde{Y}_2$ is

$$p_2 = \mathrm{sign}(\Delta x_2)\sqrt{(\tilde{u}_2)^2 - (p_1)^2} \quad . \tag{38}$$

The remaining second partial travel–time derivatives follow from equations

$$\tilde{u}_2 U_1 = T_{11} p_1 + T_{12} p_2 \ , \quad \tilde{u}_2 U_2 = T_{12} p_1 + T_{22} p_2 \tag{39}$$

and are given by

$$T_{12} = (\tilde{u}_2 U_1 - p_1 T_{11})(p_2)^{-1} \ , \quad T_{22} = (\tilde{u}_2 U_2 - p_1 T_{12})(p_2)^{-1} \ . \tag{40}$$

Now the slowness vector at Y_2 may be approximated by the Taylor expansion,

$$\bar{p}_{12} = p_1 - T_{11}\tilde{x}_{12} + T_{12}\Delta x_2 \ , \quad \bar{p}_{22} = p_2 - T_{12}\tilde{x}_{12} + T_{22}\Delta x_2 \ . \tag{41}$$

Finally, the slowness vector at gridpoint Y_2 is renormalized,

$$p_{12} = u_2\,\bar{p}_{12}/\sqrt{(\bar{p}_{12})^2 + (\bar{p}_{22})^2} \ , \quad p_{22} = u_2\,\bar{p}_{22}/\sqrt{(\bar{p}_{12})^2 + (\bar{p}_{22})^2} \ . \tag{42}$$

3.1.6 Travel–time evaluation

Then, equation

$$\tau_2 = \tau + \tfrac{1}{2}\left\{-[p_1 + p_{12}]\tilde{x}_{12} + [p_2 + p_{22}]\Delta x_2\right\} \tag{43}$$

may be used to calculate travel time at updated gridpoint Y_2 using averaged slowness vectors at points $\tilde{Y}_2$ and Y_2. Equation (43) is accurate up to the second–order terms in the Taylor expansion of travel time.

3.2 Propagation of type (b) : diffractions at gridpoints

The travel time between point $\mathbf{x} = (x_1, 0)$ and $\mathbf{x}_2 = (0, \Delta x_2)$ is approximately

$$\Delta\tau = \frac{u(x_1, 0) + u_2}{2}\sqrt{[x_1]^2 + [\Delta x_2]^2} \quad . \tag{44}$$

The accuracy of (44) is very similar (although not exactly the same) as of (43). The corresponding slowness vector at Y_2 may be approximated as

$$(\bar{p}_{12}, \bar{p}_{22}) = \frac{u(x_1, 0) + u_2}{2}\,\frac{(-x_1, \Delta x_2)}{\sqrt{[x_1]^2 + [\Delta x_2]^2}} + \frac{1}{2}(U_1, U_2)\sqrt{[x_1]^2 + [\Delta x_2]^2} \tag{45}$$

and renormalized according to (42). For propagation from Y_0 to Y_2, $x_1 = 0$ in (44) and (45), and for propagation from Y_1 to Y_2, $x_1 = \Delta x_1$ in (44) and (45).

4 Accuracy estimations

We denote by U_{11}, $U_{12} = U_{21}$, and U_{22} the second partial derivatives of slowness u in the local coordinates, by T_{IJK} the third partial derivatives of travel time in the local coordinates, and by $\mathbf{E} = (E_1, E_2)$ the vector from the backprojection $\tilde{Y}_2$ to updated point Y_2,

$$\mathbf{E} = (-\tilde{x}_{12}, \Delta x_2) \quad , \tag{46}$$

either in the local coordinates.

4.1 Error of slowness–vector extrapolation

Error of the slowness–vector extrapolation (34) due to neglected third partial travel–time derivatives is

$$\Delta \bar{p}_I = -\tfrac{1}{2} T_{IJK} E_J E_K \tag{47}$$

per a single extrapolation. By renormalization (42), the error is projected onto the wavefront tangent plane,

$$\Delta p_I = \left(\delta_{IM} - \frac{E_I E_M}{|\mathbf{E}|^2} \right) \Delta \bar{p}_M \quad . \tag{48}$$

The error is then accumulated during repeated interpolations (14) with a quotient q, see (16),

$$\delta p_1 = \Delta p_1 \, (1 - q)^{-1} = -\Delta p_1 \, (6 a_0 a_1)^{-1} \quad . \tag{49}$$

The relative travel–time error due to this accumulated slowness–vector error is, see (12),

$$\rho\tau_{(1)} = a_0 a_1 (a_0 + a_1) \, \delta p_1 \, \Delta x_1 \, (u |\mathbf{E}|)^{-1} = -(a_0 + a_1) \, \Delta p_1 \, \Delta x_1 \, (6 u |\mathbf{E}|)^{-1} \quad , \tag{50}$$

that may be rewritten, using (13), (47), and (48), in the form of

$$\rho\tau_{(1)} = \frac{2\tilde{x}_{12} - \Delta x_1}{12 \, u} \left(\delta_{1M} - \frac{E_1 E_M}{|\mathbf{E}|^2} \right) \frac{T_{IJK} E_J E_K}{|\mathbf{E}|} \quad . \tag{51}$$

4.2 Error of quadratic travel–time approximation

Error of quadratic travel time approximation (17), caused by a neglected third partial travel–time derivative, is

$$\rho\tau_{(2)} = -\frac{T_{111} \, \tilde{x}_{12} (\tilde{x}_{12} - \Delta x_1)(2\tilde{x}_{12} - \Delta x_1)}{12 \, u \, |\mathbf{E}|} \quad . \tag{52}$$

4.3 Error of linear slowness interpolation

The error of linear slowness interpolation (6) at point $\tilde{Y}_2$ is

$$\delta \tilde{u}_2 = \tfrac{1}{2} U_{11} \tilde{x}_{12} (\Delta x_1 - \tilde{x}_{12}) \quad . \tag{53}$$

This slowness error introduces the additional relative error of

$$\rho\tau_{(3)} = \frac{1}{4} \frac{U_{11}}{u} \tilde{x}_{12} (\Delta x_1 - \tilde{x}_{12}) \tag{54}$$

in travel–time approximation (43).

4.4 Error of travel–time evaluation

The relative error of travel–time approximation (43) is

$$\rho\tau_{(4)} = \frac{T_{IJK} E_I E_J E_K}{12 u |\mathbf{E}|} = \frac{|\mathbf{E}|^2}{12 u^4} \, T_{IJK} \, p_I \, p_J \, p_K \quad . \tag{55}$$

Differentiating the eikonal equation, we arrive at

$$T_{IJK} \, p_I \, p_J \, p_K = u \, p_J \, U_{JK} \, p_K + p_J \, U_J U_K \, p_K - u^2 U_M U_M \quad . \tag{56}$$

Then equation (55) may also be expressed in the equivalent form of

$$\rho\tau_{(4)} = \frac{1}{12 u^2} E_J \left(u U_{JK} + U_J U_K - U_M U_M \delta_{JK} \right) E_K \quad . \tag{57}$$

4.5 Estimation of the total relative travel–time error

We assume that the quadratic travel time approximation of Section 3.1.3.2 is used, and error (51) need not be considered. The extremal value

$$\rho\tau_{(2)\mathrm{MAX}} \approx -\frac{T_{111}}{12\, u} \frac{(\Delta x_1)^2}{10.6} \tag{58}$$

of error (52), reached for $\tilde{x}_{12} = 0.206 \Delta x_1$, is smaller than 14% of the extremal value of term

$$\frac{T_{111} E_1 E_1 E_1}{12 u |\mathbf{E}|} \tag{59}$$

appearing as a part of (55). This directs us to neglect error (52) in comparison with error (55), and to consider only errors (54) and (57).

Taking into account that

$$\Delta x_1 = h \, \mathrm{sign}(\tilde{x}_{12}) \quad , \tag{60}$$

see (10) and (11), the total relative error

$$\rho\tau = \rho\tau_{(3)} + \rho\tau_{(4)} \tag{61}$$

may be expressed in the form of

$$\rho\tau = \frac{1}{12u^2}\left[(-2uU_{11} - U_2U_2)|\tilde{x}_{12}|^2\right.$$

$$\left.+[3uU_{11} - 2(uU_{12} + U_1U_2)\,\mathrm{sign}(\Delta x_1 \Delta x_2)]|\tilde{x}_{12}|\,h + (uU_{22} - U_1U_1)h^2\right] \ . \quad (62)$$

This function $\rho\tau = \rho\tau(|\tilde{x}_{12}|)$ has, in interval $|\tilde{x}_{12}| \in < 0, h >$, 2 to 3 local extremes,

$$\rho\tau(0) = \frac{h^2}{12u^2}[uU_{22} - U_1U_1] \quad , \quad (63)$$

$$\rho\tau(h) = \frac{h^2}{12u^2}\left[uU_{11} - U_2U_2 - 2(uU_{12} + U_1U_2)\,\mathrm{sign}(\Delta x_1 \Delta x_2) + uU_{22} - U_1U_1\right] \ , \quad (64)$$

and if

$$|\tilde{x}_{12}| = \frac{3uU_{11} - 2(uU_{12} + U_1U_2)\,\mathrm{sign}(\Delta x_1 \Delta x_2)}{2(2uU_{11} + U_2U_2)}\ h \quad (65)$$

belongs to interval $|\tilde{x}_{12}| \in < 0, h >$, also

$$\rho\tau(|\tilde{x}_{12}|) = \frac{h^2}{12u^2}\left\{\frac{[3uU_{11} - 2(uU_{12} + U_1U_2)\,\mathrm{sign}(\Delta x_1 \Delta x_2)]^2}{4(2uU_{11} + U_2U_2)} + uU_{22} - U_1U_1\right\}$$

$$= \frac{h^2}{12u^2}\left\{(2uU_{11} + U_2U_2)\frac{|\tilde{x}_{12}|^2}{h^2} + uU_{22} - U_1U_1\right\} \quad . \quad (66)$$

In 2-D, the global Cartesian coordinates may be either shifted local coordinates (x_1, x_2), or shifted exchanged local coordinates (x_2, x_1), see Section 3.1. Then equation (62) for the relative travel–time error may be rewritten in global Cartesian coordinates: for $|p_1| \leq |p_2|$,

$$\rho\tau = \frac{h^2}{12u^2}\left[(-2uU_{11} - U_2U_2)\left|\frac{p_1}{p_2}\right|^2\right.$$

$$\left.+[3uU_{11} + 2(uU_{12} + U_1U_2)\,\mathrm{sign}(p_1p_2)]\ \left|\frac{p_1}{p_2}\right| + (uU_{22} - U_1U_1)\right] \quad . \quad (67)$$

and for $|p_2| \leq |p_1|$,

$$\rho\tau = \frac{h^2}{12u^2}\left[(-2uU_{22} - U_1U_1)\left|\frac{p_2}{p_1}\right|^2\right.$$

$$\left.+[3uU_{22} + 2(uU_{12} + U_1U_2)\,\mathrm{sign}(p_1p_2)]\ \left|\frac{p_2}{p_1}\right| + (uU_{11} - U_2U_2)\right] \quad . \quad (68)$$

where the second partial slowness derivatives are considered, unlike before, in global Cartesian coordinates.

5 Comparison with the first–order methods of grid travel–time tracing and network ray tracing

5.1 Comparison with network shortest–path ray tracing

We denote by U_+ the most positive and by U_- the most negative eigenvalue of matrix

$$uU_{JK} - \tfrac{1}{2}U_J U_K + \tfrac{1}{2}U_M U_M \delta_{JK} \quad . \tag{69}$$

The maximum relative travel–time error of the network shortest–path ray tracing according to Klimeš & Kvasnička (1994) in a smooth 2-D medium is then

$$\rho\tau_{\rm NET} \approx \sqrt{\frac{h^2}{24u^2}\max(U_+, \tfrac{1}{2}|U_-|)} \quad . \tag{70}$$

Comparing equations (69) and (70) with equations (57), (67), and (68), the relation between relative errors of grid travel–time tracing of the second order and network shortest–path ray tracing may be estimated very roughly as

$$|\rho\tau| \approx 2(\rho\tau_{\rm NET})^2 \quad . \tag{71}$$

5.2 Comparison with first–order grid travel–time tracing

Let us now estimate the relative travel–time error of the first–order grid travel–time tracing method by Podvin & Lecomte (1991). Assume that travel times $\tau_\pm$ at 2 gridpoints with coordinates $(\pm\frac{1}{2}h, -\frac{1}{2}h)$ are known and travel time τ at gridpoint with coordinates $(\frac{1}{2}h, \frac{1}{2}h)$ is to be calculated. The Taylor expansions of the travel times at point $(0,0)$ are

$$\tau_\pm = T \pm p_1\tfrac{1}{2}h - p_2\tfrac{1}{2}h + \tfrac{1}{2}(T_{11}+T_{22})(\tfrac{1}{2}h)^2 \mp T_{12}(\tfrac{1}{2}h)^2 \quad , \tag{72}$$

$$\tau = T + p_1\tfrac{1}{2}h + p_2\tfrac{1}{2}h + \tfrac{1}{2}(T_{11}+T_{22})(\tfrac{1}{2}h)^2 + T_{12}(\tfrac{1}{2}h)^2 \quad . \tag{73}$$

The calculated travel time at point $(\frac{1}{2}h, \frac{1}{2}h)$ is

$$\begin{aligned}\tau_{\rm FIRST} = \tau_+ + h\sqrt{u^2 - \left(\frac{\tau_+ - \tau_-}{h}\right)^2} &= \tau_+ + h\sqrt{u^2 - (p_1 - T_{12}\tfrac{1}{2}h)^2} \\ \approx \tau_+ + h\sqrt{(p_2)^2 + p_1 T_{12} h} &\approx \tau_+ + h\,(p_2 + \tfrac{1}{2}\frac{p_1}{p_2}T_{12}h) \quad . \end{aligned}\tag{74}$$

Here $u = \sqrt{(p_1)^2 + (p_2)^2}$ is the slowness at the centre $(0,0)$ of the cell. The absolute error is then

$$\delta\tau_{\rm FIRST} = \tau_{\rm FIRST} - \tau \approx -p_2 h - 2T_{12}(\tfrac{1}{2}h)^2 + p_2 h + \tfrac{1}{2}\frac{p_1}{p_2}T_{12}h^2 = -\tfrac{1}{2}(1 - \frac{p_1}{p_2})T_{12}h^2 \ . \tag{75}$$

The corresponding relative travel–time error is

$$\rho\tau_{\rm FIRST} = \delta\tau_{\rm FIRST}\frac{p_2}{u^2 h} \approx -\frac{h}{2u^2}(p_2 - p_1)T_{12} \quad . \tag{76}$$

This result is valid in local coordinates chosen in such a way that both the local slowness vector components p_1 and p_2 are positive and $p_1 \le p_2$. For instance, for the propagation in the direction of the x_2 axis, $p_2 - p_1 = u$ and $T_{12} = U_1$. This error may roughly be twice greater than that of network shortest–path ray tracing, and the relation between relative errors of grid travel–time tracing of the second and first orders may be estimated very roughly as

$$\rho\tau \approx \tfrac{1}{2}(\rho\tau_{\rm FIRST})^2 \quad . \tag{77}$$

6 Numerical example: Constant velocity gradient model

In 1-D constant velocity gradient model, velocity increases linearly with depth. The velocity ratio between top and bottom is 1:3. The source is situated at the left–hand top corner of a vertical square in which the first–arrival travel times are calculated. The model has been used by Podvin & Lecomte (1991) to study the accuracy of their first–order grid travel–time tracing method, and then by Klimeš & Kvasnička (1994) to demonstrate the accuracy of network shortest–path ray tracing.

The 2-D grid of 201×201 nodes (200×200 cells) was chosen to cover the square, like in two papers referred above. The second–order grid travel–time tracing method proposed in this paper has been compared with the results of network shortest–path ray tracing. The analytical solution has been subtracted from the first–arrival travel times calculated at the gridpoints by both the methods to yield the absolute travel–time errors.

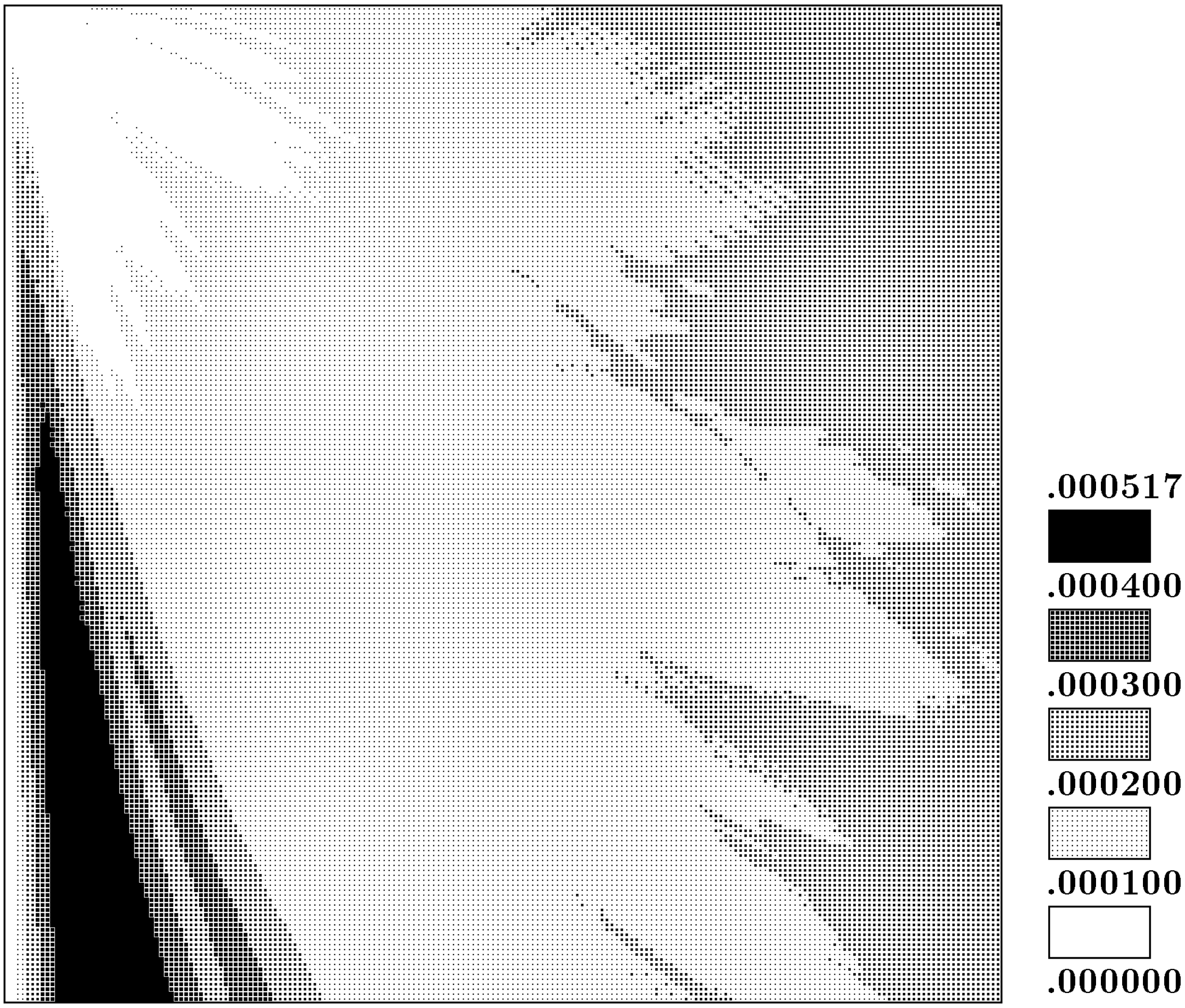

Figure 3a. Constant velocity gradient model. Travel–time errors of the network shortest–path ray tracing according to Klimeš & Kvasnička (1994). The absolute errors of the first–arrival travel time are scaled in seconds. Program `NET` version `2.00`.

For comparison, the absolute network travel–time errors are shown in Fig. 3a. Let us note that the errors of the first–order grid travel–time tracing method by Podvin & Lecomte (1991) are roughly twice greater, see Klimeš, Kvasnička & Červený (1994).

The absolute travel–time errors of the second–order grid travel–time tracing methods proposed in this paper are shown in Figures 3b and 3c. The two methods differ in approach to the travel–time interpolation between adjacent gridpoints. Figure 3b corresponds to the cubic travel–time interpolation between adjacent gridpoints, described in Sections 3.1.3.1 and 3.1.4.1. Figure 3c corresponds to the quadratic approximation of travel time between adjacent gridpoints, described in Sections 3.1.3.2 and 3.1.4.2. The latter approach is now preferred by the author because of probably better stability achieved by elimination of accumulating differences between calculated slowness vectors and derivatives of calculated travel time.

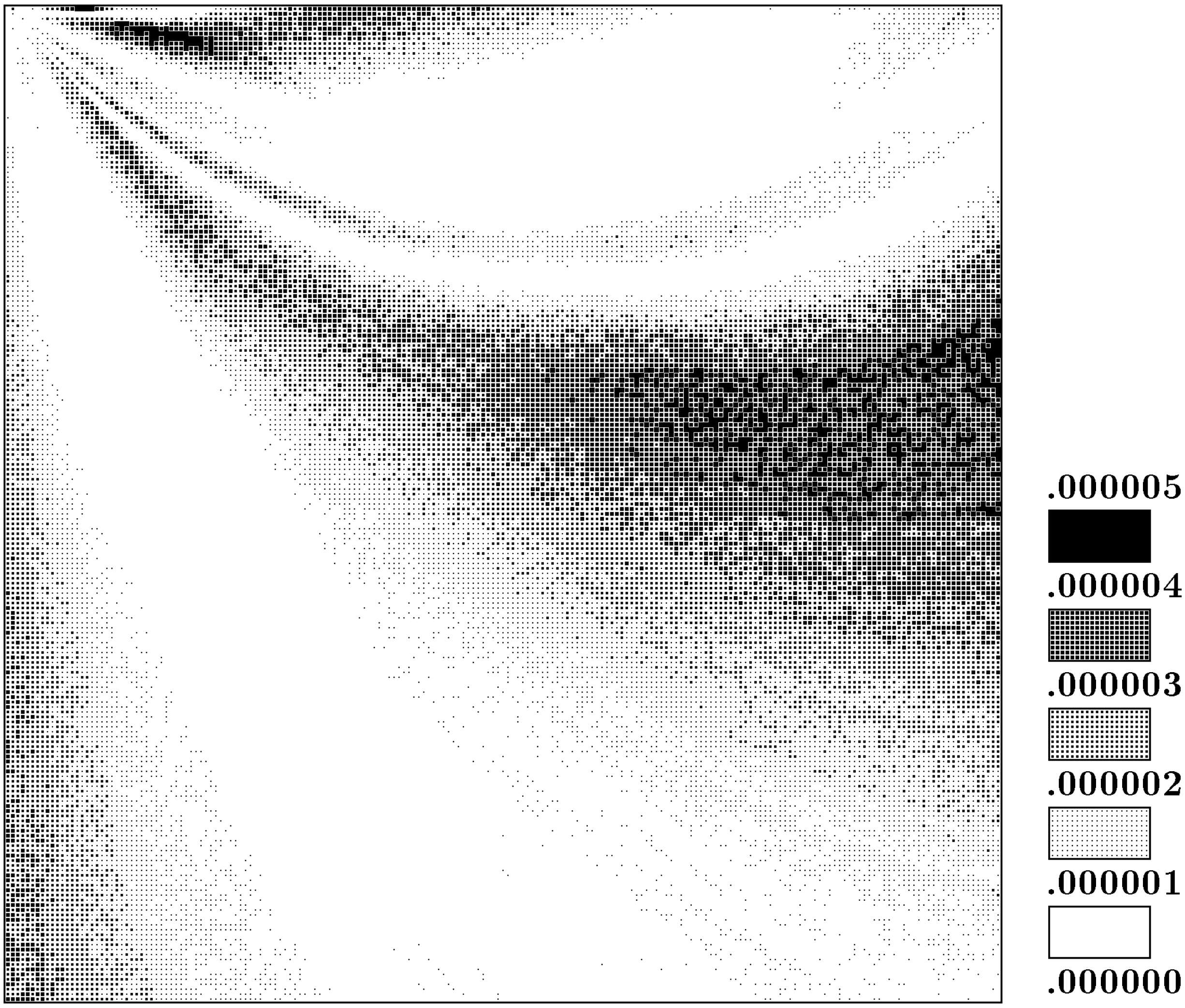

Figure 3b. Constant velocity gradient model. Travel–time errors of the second–order grid travel–time tracing according to this paper, with cubic travel–time interpolation along gridlines. The errors are scaled in seconds. Let us note that the error distribution has somewhat random nature since the rounding errors are of the order of 0.000001 s, and the numerical accuracy of evaluating the reference analytical travel times is, at some gridpoints, even worse. Program `TTT-2D` version `0.06`.

Let us note that the errors in Figures 3b and 3c contain somewhat random component due to rounding errors: both the numerically traced and analytical travel times have been rounded to 6 decimal places before subtracting. Moreover, the numerical errors of evaluating analytical travel times may reach at some points absolute error of 0.000002 s or even more. For example, the error at the second gridpoint of the top row in Figures 3b and 3c is due to the analytical solution, grid travel–time tracing is much more accurate there.

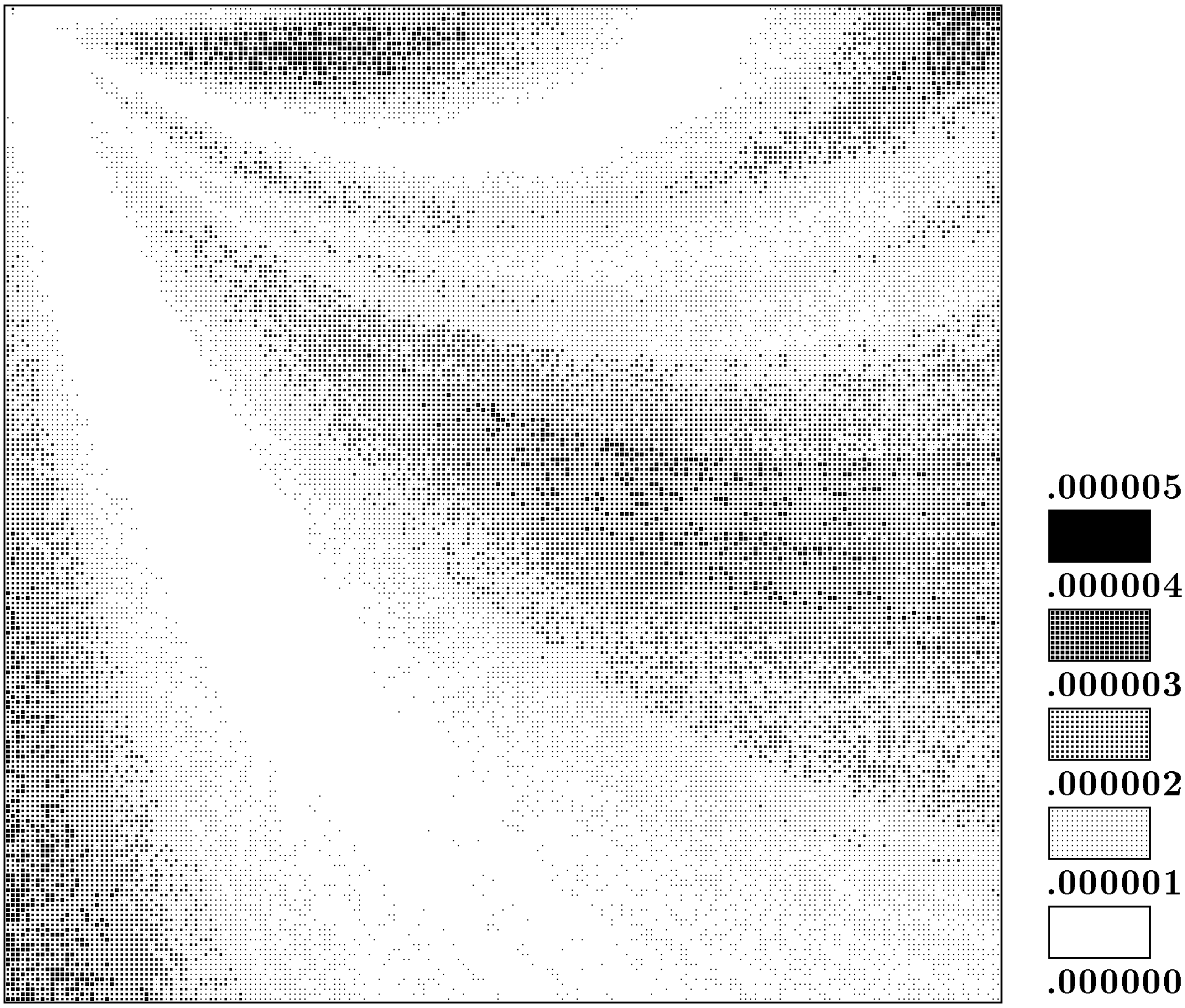

Figure 3c. Constant velocity gradient model. Travel–time errors of the second–order grid travel–time tracing according to this paper, with quadratic travel–time approximation along gridlines. The errors are scaled in seconds. Let us note that the error distribution has somewhat random nature since the rounding errors are of the order of 0.000001 s, and the numerical accuracy of evaluating the reference analytical travel times is, at some gridpoints, even worse. Program `TTT-2D` version `0.07`.

7 Structural interfaces

The above algorithm should not be used, without modification, at structural interfaces of the first order at least for two reasons:

(α) The method is designed for a smooth medium with heterogeneities sufficiently larger then the grid interval, and is unstable at presence of considerable slowness (or velocity) contrasts.

(β) The travel–time error due to the structural interfaces is of the order of h, that is inadequate to the method of the second–order accuracy.

Thus, in addition to gridded slownesses (or velocities), also information on the positions and slopes of structural interfaces has to be available to the grid travel–time tracing program. The algorithm has then be extended to take the Snell law into account at the interfaces.

8 Conclusions

The proposed method of grid travel–time tracing looks very promising to fulfill all practical requirements on the accuracy and speed of the first–arrival travel–time calculation. Its accuracy is about the square of the accuracy of the contemporary first–order methods. For example, if the relative travel–time error of a first–order method is of the order of 1%, the error of the proposed second–order method should be of the order of 0.01%.

The algorithm enables to evaluate, in addition to calculated first–arrival travel times, also approximately their errors in order to check the accuracy of calculation.

In order to be generally applicable, the proposed algorithm has to be *extended* to *3-D*, and to *block models with curved structural interfaces* of the first order.

The method may be supplemented with an a posteriori *ray tracer*. The ray tracer may take advantage of the slowness vectors determined by the algorithm.

Acknowledgements

The research has been supported by the Grant Agency of Charles University under the contract 8/94, by the Grant Agency of Czech Republic under the contract 205/95/1465, by the Commission of European Union in the framework of the JOULE II project "Integrated Structural Imaging of Seismic Data", and by Institut Français du Pétrole, Shell Research B.V., and Japan National Oil Corporation within the consortium project "Seismic Waves in Complex 3-D Structures".

References

Klimeš, L., Kvasnička, M. & Červený, V. (1994): Grid computations of rays and travel times. In: Seismic waves in Complex 3-D structures. Report 1. Dept. Geophys., Faculty Math. Phys., Charles Univ., Prague, pp.85–114.

Klimeš, L. & Kvasnička, M. (1994): 3-D network ray tracing. Geophys. J. int., **116**, 726–738.

Moser, T-J. (1989): Efficient seismic ray tracing using graph theory. 59th annual int. meeting, Soc. Exploration Geophys., Expanded Abstracts, 1106–1108.

Moser, T-J. (1991): Shortest path calculation of seismic rays. Geophysics, **56**, 59–67.

Nakanishi, I. & Yamaguchi, K. (1986): A numerical experiment on nonlinear image reconstruction from first–arrival times for two–dimensional island arc structure. J. Phys. Earth, **34**, 195–201.

Podvin, P. & Lecomte, I. (1991): Finite difference computation of traveltimes in very contrasted velocity models: a massively parallel approach and its associated tools. Geophys. J. int., **105**, 271–284.

Saito, H. (1989): Traveltimes and raypaths of first arrival seismic waves: computation method based on Huygens' principle. 59th annual int. meeting, Soc. Exploration Geophys., Expanded Abstracts, 244–247.

Saito, H. (1990): 3-D ray–tracing method based on Huygens' principle. 60th annual int. meeting, Soc. Exploration Geophys., Expanded Abstracts, 1024–1027.

Vidale, J. (1988): Finite–difference calculation of travel times. Bull. seismol. Soc. Am., **78**, 2062–2076.