Finite differencing with uneven spatial sampling

Joe Dellinger and *Francis Muir*

INTRODUCTION

Seismic data is often not recorded on a regular grid. This is especially likely to be the case in the cross-line direction in marine 3-D surveys, where various navigational and practical problems insure that the boat will not travel exactly parallel to previous traverses. This poses special design problems for migration. A common solution is to map the data onto a regular grid using some interpolation scheme, and, in case of migration, another popular solution is to employ integral methods, which, superficially, seem to steer around the problem. While these methods have served us well enough, it is still possible to use differencing schemes on the original, irregular grid.

In this paper we show how to extend standard (ω, x, z) one-way wave equation finite differencing in the case that the spatial axis is irregularly sampled. Doing this so as to guarantee unconditional stability and to minimize the generation of artifacts requires careful construction of the finite difference operators. This is particularly important because, as a practical matter, grid irregularities must be on the scale of the finite difference grid. The math is somewhat more complex than in the regular case, but, in the end, the finite differencing star is the same size and shape as before.

Our scheme appears to be quite robust against extreme variations in the grid spacing, but spurious reflections must occur when the spacing exceeds the Nyquist limit. Stability has also not yet been proven for the irregular extension of the so-called "1/6 trick". Solutions for these remaining problems are still a subject of active research.

DIFFERENTIAL EQUATIONS

We begin by carefully developing Muir's one-way equations, being extra careful to keep track of both velocity and impedance and to not assume that these are constant.

Start with a scalar wave equation in (ω, x, z) space:

$$(\nabla \cdot \frac{1}{\rho} \nabla + \frac{\omega^2}{\mathrm{K}}) P = 0,$$

where K is the bulk modulus, ρ is the density, and P is pressure. (We could just as well have used a slightly different form of the equation for a characteristic scalar displacement field.)

We wish to remove the bulk modulus from the ω term, but still preserve the symmetry of the second derivative term, so that when we discretize later it will simplify the stability proof. To do this we pre- and post-multiply by $\sqrt{\mathrm{K}}$, and change the field variable appropriately, obtaining

$$(\sqrt{\mathrm{K}} \nabla \cdot \frac{1}{\rho} \nabla \sqrt{\mathrm{K}} + \omega^2) \frac{P}{\sqrt{\mathrm{K}}} = 0.$$

Recasting this in terms of velocity $\mathrm{V} = \sqrt{\frac{\mathrm{K}}{\rho}}$ and impedance $\mathrm{I} = \sqrt{\mathrm{K}\rho}$ we obtain

$$[\sqrt{\mathrm{IV}}\ \nabla \cdot \frac{\mathrm{V}}{\mathrm{I}}\ \nabla\ \sqrt{\mathrm{IV}} + \omega^2]\ \frac{P}{\sqrt{\mathrm{IV}}} = 0. \tag{1}$$

Our field variable is now $\frac{P}{\sqrt{\mathrm{IV}}}$, which has units of energy.

One way equations

If I, the impedance, is now chosen to be independent of z within a layer, we can rewrite equation 1 as

$$[(\sqrt{\mathrm{V}} \frac{\partial}{\partial z} \sqrt{\mathrm{V}})^2 + \sqrt{\mathrm{VI}} \frac{\partial}{\partial x} \frac{\mathrm{V}}{\mathrm{I}} \frac{\partial}{\partial x} \sqrt{\mathrm{VI}} + \omega^2] \frac{P}{\sqrt{\mathrm{VI}}} = 0.$$

We can now take the square root of the z derivative, and get a one-way wave equation by taking a specific sign.

$$[\sqrt{\mathrm{V}} \frac{\partial}{\partial z} \sqrt{\mathrm{V}} - \sqrt{-(\sqrt{\mathrm{VI}} \frac{\partial}{\partial x} \frac{\mathrm{V}}{\mathrm{I}} \frac{\partial}{\partial x} \sqrt{\mathrm{VI}} + \omega^2)}] \frac{P}{\sqrt{\mathrm{VI}}} = 0 \tag{2}$$

A fact which is usually glossed over at this point is that our field variable $\frac{P}{\sqrt{\mathrm{VI}}}$ has also been changed. Before it was the total energy field; the square root has segregated the up- and down-going components. "Field data" technically should be processed to separate out the up-going component before migration. (See Aki and Richards, volume I, for more on this.)

The square root is expanded into a continued fraction, using the recurrence

$$R_0 = W, \qquad R_n = W + X(W + R_{n-1})^{-1}.$$

If X and W commute, then this latter expression will converge to $\sqrt{W^2 + X}$, under certain positivity constraints to be later discussed. We will set

$$W = i\omega$$

and

$$X = -\sqrt{\mathrm{VI}}\frac{\partial}{\partial x}\frac{\mathrm{V}}{\mathrm{I}}\frac{\partial}{\partial x}\sqrt{\mathrm{VI}}. \tag{3}$$

This gives us a family of approximate one-way equations. We now need to find usable finite difference implementations of these.

FINITE DIFFERENCING

The problem is now to represent $\frac{\partial}{\partial x}$ when our finite differencing grid is irregular. We can always cascade two first derivatives together to get a second derivative.

We are given a set of data points. The standard 2 point first derivative is equivalent to connecting each pair of adjacent points by a straight line. The slope of this line is considered to approximate the first derivative of the function everywhere between the pair of points. It is obviously a bad approximation near the ends of this interval, and the best in the middle. If we are free to pick the grid upon which the first derivative is to be represented, it is best to choose the midpoint.

This is illustrated in figure 1. The original data is at the top, with the sample points marked with a "0". The middle curve shows the exact continuous and the 2 point finite difference first derivative. The new gridpoints are halfway between the old ones. Despite the quite irregular spacing, the first derivative is accurate.

We can simply take another derivative just as before and get a new set of gridpoints, shown in the figure with an "X". In general, these will not be the same as the original ones. Since we are approximating the first derivative by a straight line between each two gridpoints, if it is required that we end up back on our original grid, we can simply shift the point over to the position of the original gridpoint in that interval. These points are marked with a "2". There is a loss in accuracy in doing this, which is worse the more irregular the sampling is.

It would be nice to let the gridpoints drift, which would both keep the extra accuracy and slowly "regularize" the sample spacing. Unfortunately this also creates bookkeeping problems. Furthermore, in the wave equation you must add undifferentiated and differentiated terms, which would then not be on the same grid. There may be ways around these problems, but for now we will keep the original grid intact.

Finite differencing with a one-way wave equation

We now wish to find an appropriate finite difference representation of "X" from equation 3.

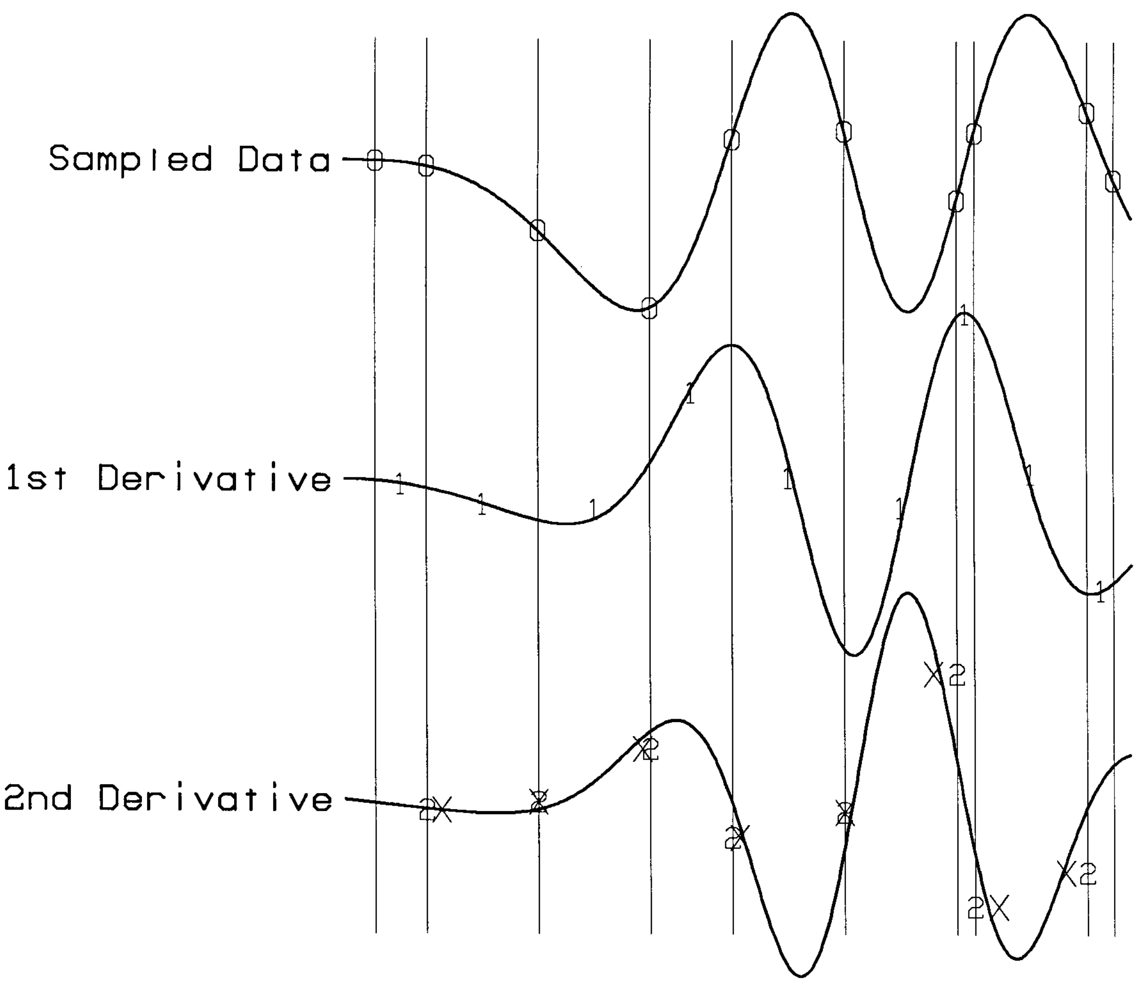

Figure 1: Taking derivatives numerically with irregular sampling. The vertical lines show where the samples are. The grid points for the first derivative are midway between the original grid-points. If we simply repeat the process to get the second derivative, we again get new grid points midway between the old. This is shown by the X's. If instead we would like to end up back on the original coordinate grid, we get the points marked with a 2.

Let B be the bidiagonal matrix

$$\begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & \ddots & -1 \\ & & & 1 \end{bmatrix}.$$

This matrix does the real work of taking a derivative. It also has the unwanted effect of shifting the gridpoints half a step to the right. In order not to end up with the grid shifted over a whole sample after taking two derivatives, we first use this matrix and then the negative of its transpose, which also differences as does B itself but shifts the grid in the opposite direction, undoing the shift. This shift is really an artifact of our matrix notation, but comes in useful when we want to show that our operator is positive semidefinite.

The matrix B alone does not take a derivative. After subtracting two adjacent samples we must then normalize by the length of the interval between them. In matrix notation, this is can be done by putting the appropriate normalization factors into a diagonal matrix (which is Δx^{-1}) and premultiplying this onto B.

The resulting finite difference representation of X is then

$$\sqrt{\mathrm{VI}}\ \Delta x^{-1}\ B^T\ \frac{\mathrm{V}}{\mathrm{I}}\ \Delta x'^{-1}\ B\ \sqrt{\mathrm{VI}}. \tag{4}$$

Unfortunately, Δx^{-1} occurs to the left of both B matrices. In order for our final differencing method to be stable, this matrix must be positive semidefinite. And here is the central problem, because Δx^{-1} does not commute with B or B^T, and Δx^{-1} is not even the same as $\Delta x'^{-1}$. We will have to do some manipulations involving the entire equation to regain a positive semidefinite form.

45° equation modeling

The 45° approximation to equation 2 is

$$[\sqrt{\mathrm{V}}\frac{\partial}{\partial z}\sqrt{\mathrm{V}} - i\omega + X(2i\omega + X(2i\omega)^{-1})^{-1}]\frac{P}{\sqrt{\mathrm{VI}}} = 0. \tag{5}$$

We need to replace X by the positive semidefinite form

$$\left[(\frac{\mathrm{V}}{\mathrm{I}})^{\frac{1}{2}}\ \Delta x'^{-\frac{1}{2}}\ B\,\Delta x^{-\frac{1}{2}}\ (\mathrm{VI})^{\frac{1}{2}}\right]^T \left[(\frac{\mathrm{V}}{\mathrm{I}})^{\frac{1}{2}}\ \Delta x'^{-\frac{1}{2}}\ B\,\Delta x^{-\frac{1}{2}}\ (\mathrm{VI})^{\frac{1}{2}}\right]$$

which is

$$\Delta x^{+\frac{1}{2}} X \Delta x^{-\frac{1}{2}}.$$

Using the fact that $AB^{-1} = ACC^{-1}B^{-1} = AC(BC)^{-1}$, and also that Δx^{-1} commutes with everything except X, we can easily get equation 5 into the required form

$$[\sqrt{\mathrm{V}}\frac{\partial}{\partial z}\sqrt{\mathrm{V}} - i\omega + \Delta x^{+\frac{1}{2}} X \Delta x^{-\frac{1}{2}} (2i\omega + \Delta x^{+\frac{1}{2}} X \Delta x^{-\frac{1}{2}} (2i\omega)^{-1})^{-1}] \frac{P}{\sqrt{\Delta x^{-1}\,\mathrm{VI}}} = 0.$$

This is the "bullet proof" form we desire. Note that the units of

$$\|\frac{P}{\sqrt{\Delta x^{-1}\,\mathrm{VI}}}\|^2$$

are those of energy flux. This is the amount of energy flowing through the region of space represented by the sample point in unit time.

The correction for the sample spacing in the field variable is quite important. Initial data given in terms of pressure must be converted into field variable units, and then converted back for output.

AN EXAMPLE

In figure 2 is the central portion of one frame of a movie showing waves propagating from a point source. The 15° equation was used. Three especially large gaps in the grid are apparent. The grid outside the gaps is also very irregular. At the first gap, where the dips are relatively low, there is no reflection apparent. For the higher dips found on the right side of the plot, reflections from irregularities in the grid spacing are higher in amplitude. In general, the modeling will break down in regions where there is significant energy beyond the spatial nyquist frequency.

The problem is that aliasing of the operator aliases the dispersion relation. Differing grid-point spacings will have differing spatial nyquists, which will change the position of the aliased portions of the dispersion relation. A change in the sample spacing will cause the aliased portions of the dispersion relations for adjacent operators to mismatch, causing reflections.

CONCLUSION

In general, any differential equation can be solved numerically by integral or differential schemes. Just as integral methods can be used with irregularly sampled data, so can difference methods. This new "bullet-proof" scheme should prove particularly useful in the migration of marine cross-line data.

ACKNOWLEDGMENTS

We wish to thank Jon Claerbout for suggesting this research topic, and for his continuing interest in the project.

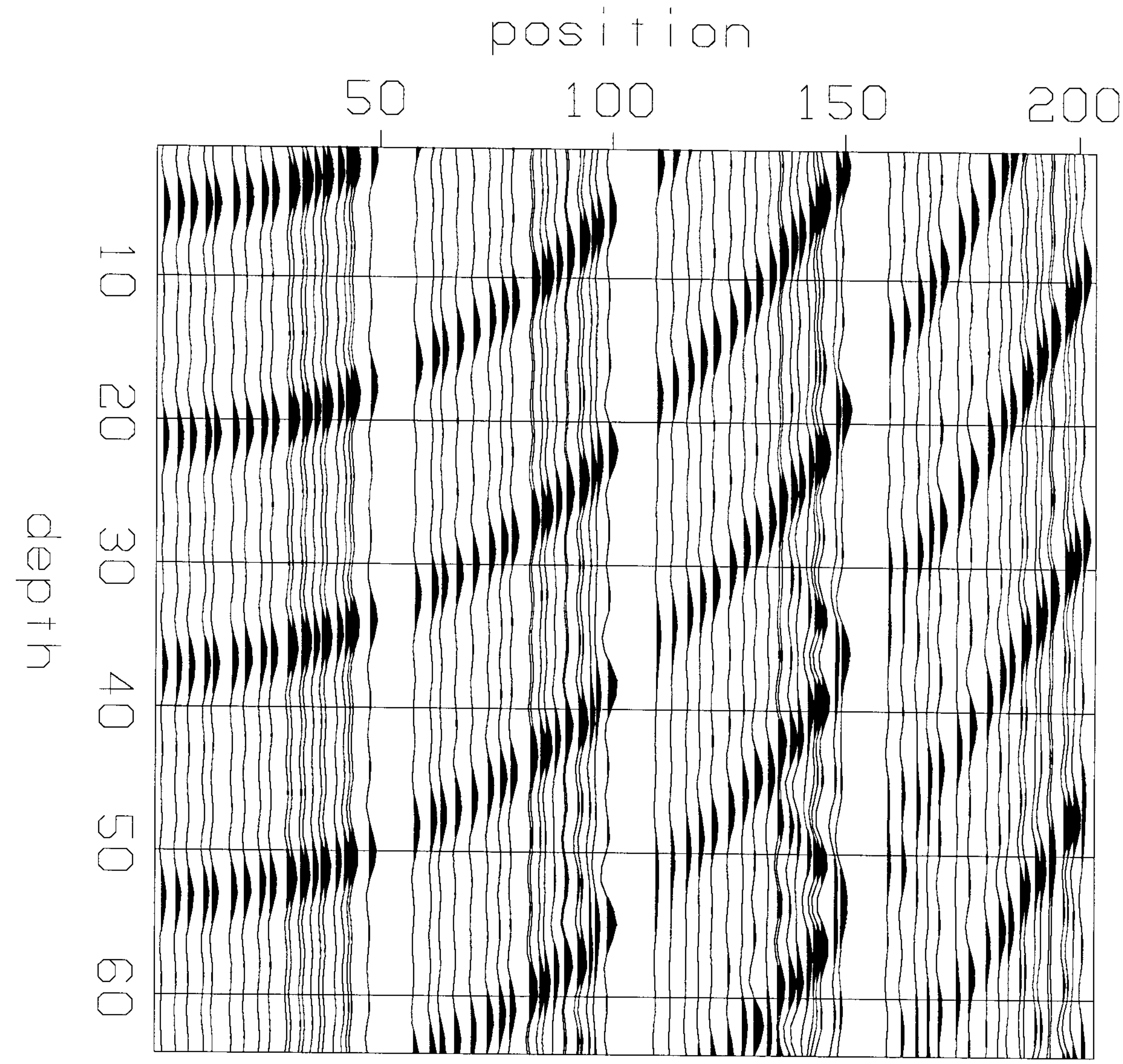

Figure 2: Waves propagating from a point source using 15° finite differencing on a very irregularly sampled grid. The depth axis runs from top to bottom, and the irregularly sampled space axis runs from left to right. This can be viewed as a snapshot of the wavefield at an instant in time.

Weird code, part 3

Yet another weird program which came over the net recently. Yes, it is valid C language, and yes, it supposedly compiles and runs.

---

3. The strangest appearing program:
(submitted by Ed Lycklama <decvax!cca!ima!ism780!ed>)

```
#define o define
#o ___o write
#o ooo (unsigned)
#o o_o_ 1
#o _o_ char
#o _oo goto
#o _oo_ read
#o o_o for
#o o_ main
#o o__ if
#o oo_ 0
#o _o(_,__,___)(void)___o(_,__,ooo(___))
#o __o (o_o_<<((o_o_<<(o_o_<<o_o_))+(o_o_<<o_o_)))+(o_o_<<(o_o_<<(o_o_<<o_o_)))
o_(){_o_ _=oo_,__,___,____[__o];_oo ______;_____:___=__o-o_o_; _______:
_o(o_o_,____,__=(_-o_o_<___?_-o_o_:___));o_o(;__;_o(o_o_,"\b",o_o_),__--);
_o(o_o_," ",o_o_);o__(--___)_oo _______;_o(o_o_,"\n",o_o_);______:o__(_=_oo_(
oo_,____,__o))_oo _____;}
```

[it looks like tty noise]