A sparse, bound-respecting parametrization of velocities

Eric Dussaud and William Symes

October 29th, 2004
Motivation

• The choice of model space to describe the background velocity is essential. In particular, the model should:
  
  – provide an accurate representation of the real medium, while being sparse, to avoid oversampling in areas where the velocity varies only slightly.

  – accommodate for explicit bounds on the velocity (e.g. stability issues in finite difference schemes).

  – be adequate for seismic processing: in practice, the velocity should be:
    
    * sampled on regular grids (e.g. finite difference schemes).
    * twice continuously differentiable (high-frequency asymptotics assumption).
Proposed solution

We propose a combination of:

- a parsimonious parametrization of the velocity field, by means of user-specified nodal values.

- a computationally efficient algorithm to smoothly approximate nodal velocity values on regular grids, for models of arbitrary dimensions.

Characteristics:

- the algorithm allows for used-defined smoothness.

- it also guarantees that bounds explicitly placed on the nodes are preserved by the corresponding approximants.
Partially Irregular Grids (PIGrids). The velocities are specified as: \( v_{ijk} \equiv v(y_i, x_{ij}, z_{ijk}) \).
One-dimensional smooth approximation

The building block of the algorithm consists of:

• Piecewise linear interpolation on a regular grid.

\[ k(x) = \begin{cases} 
\frac{2}{h} \left(1 - \frac{|x|}{h/2}\right) & \text{for } -\frac{h}{2} \leq x \leq \frac{h}{2} \\
0 & \text{otherwise} 
\end{cases} \]  

(1)

where \( h \) is the smoothing width: the larger \( h \), the wider the kernel, and the more local averaging is performed.
Explicit bounds

The scheme guarantees that the order is preserved, unlike cubic spline interpolation.
Extension to the multidimension case

• We refer to the 1-D smooth approximation scheme as the SMPL operator.

• The rest of the algorithm consists of re-arranging the data into 1-D irregular samples which can then be treated by the SMPL operator.

• Example: consider the “raw” data shown before, and suppose the output (regular) grid is described by $n_z$, $\Delta z$, $n_x$, $\Delta x$, $n_y$, $\Delta y$, i.e. the number of samples and sampling interval in depth, in-line and cross-line directions, respectively.

• First step of the algorithm: apply the SMPL operator to each “vertical well” corresponding to a point $(y_i, x_{ij})$ on the surface.
Model after one step of the algorithm
Description of the second step

This step consists of looping through the discrete $z$-axis, forming irregularly samples at each level, and interpolating onto the regular samples along the $x$-axis.
Model after two steps of the algorithm
Conclusions

• A sparse, hierarchical parametrization of the velocity which allows for a user-defined placement of nodes.

• The algorithm exploits the hierarchical structure to perform the smooth approximation efficiently, it also allows for user-controlled smoothness, and guarantees that the order is preserved.

• The algorithm is reversible, in the sense that the adjoint operator of the SMPL operator and of the whole scheme exist, thus allowing its use in a gradient-based optimization context.

• In particular, the parametrization and the algorithm have been successfully used for NMO-based differential semblance optimization (see next talk).