A taut spline-based Eulerian–Lagrangian method for solving heterogeneous transport problems
Feng Ruan*, Dennis McLaughlin^ 
^Applications Research, GeoQuest, Schlumberger, Houston, TX 77056, U.S.A. E-mail: fruan@houston.geoquest.slb.com 
^Parsons Laboratory, Dept. of Civil and Env. Engineering, MIT Cambridge, MA 02139, U.S.A. E-mail: dennism@mit.edu

Abstract

In this paper we illustrate how third order taut spline interpolation can be used with the Eulerian-Lagrangian method to solve convection dominated transport problems. The spatial interpolator is implemented in a local form which reduces the computational effort required in two or three dimensional spaces. Visual accuracy assessments are used to compare solutions obtained with linear and cubic-spline interpolations. This comparison suggests that numerical accuracy depends on the interpolator's ability to resolve sharp concentration fronts at the grid scale. We observe that solutions obtained from the taut spline interpolator are less likely to exhibit artificial oscillations than solutions obtained from a cubic spline interpolator. Taut spline interpolation appears to be a promising way to improve the solution accuracy of Eulerian-Lagrangian solvers, particularly in the presence of variable velocity fields, without greatly increasing computational effort.

Introduction

Accurate solution of the multidimensional convection-dominated transport equation can be a difficult problem, especially when the velocity field is highly variable. The Eulerian-Lagrangian method is believed by many to be better able to solve such problems than either the
Eulerian or the Lagrangian methods [Ruan and McLaughlin, 1997]. However, Eulerian-Lagrangian solutions can still exhibit large numerical errors near sharp concentration fronts. A close look at the Eulerian-Lagrangian method shows that it is an operator splitting procedure. This procedure decomposes the solution of the transport equation into one step which solves the convection operator and another step which solves the dispersion operator [Wheeler et al., 1988]. At any given time, the convection step relies on spatial interpolation of both the velocity input and the concentration solution from the previous time step. A number of studies indicate that the overall solution accuracy is crucially dependent on the accuracy of the spatial interpolator [Holly and Preissmann, 1977; Rovelstad, 1994].

Many different interpolation alternatives are available. Among these, first-order linear and second-order quadratic interpolation are the most popular choices. These two interpolators are simple to implement in two and three dimensions, but they are not shape preserving. Even without the presence of large concentration variations, linear interpolation tends to smooth solutions between data points and quadratic (or higher order) interpolation tends to create spurious oscillations. Splines are different kinds of polynomials that attempt to preserve the derivatives of the interpolated function as well as the values of this function at the data points [Rovelstad, 1994]. A spline in one dimensional space is composed of piecewise polynomials defined over segments which connect points called knots. Knot locations for the cardinal spline use in this paper coincide with the data locations. This simplifies the formulation of the spline interpolator and makes it possible to express the interpolation algorithm in a computationally efficient local form.

In this work we show how a third-order taut spline interpolator can be used to obtain an accurate and efficient Eulerian-Lagrangian transport solver. A taut spline is defined as a cubic spline in tension. In one dimension the taut spline approximation to the true solution can be conceptualized as a rubber band which is stretched between adjacent knots (or data points). When the tension is high, the taut spline approaches a set of piecewise linear segments. When the tension is small the curvature between knots increases and the taut spline approaches a traditional cubic spline. It is possible to express multidimensional spline interpolation in a tensor product form which allows an $M$-dimensional interpolator to be derived from a se-
quence of $M$ one-dimensional interpolation problems. This greatly simplifies the interpolation computations required in practical multi-dimensional transport solvers. In the following sections we summarize the Eulerian-Lagrangian solution procedure and cubic taut spline interpolation. We then discuss a two-dimensional example that compares Eulerian-Lagrangian solutions obtained from linear, quadratic, cubic spline and the taut spline interpolators. We conclude with a brief review of the method's capabilities and limitations.

**Formulation**

We consider the following solute transport equation written in non-conservative form and defined over a spatial domain $\Omega$ and time interval $J = (0, T)$:

\[
\frac{\partial c}{\partial t} + \mathbf{v} \cdot \nabla c = \nabla \cdot (D \nabla c) + s \quad (x, t) \in \Omega \times J
\]
\[
c(x, 0) = c_0(x) \quad x \in \Omega
\]
\[
c = \bar{c}(x, t) \quad (x, t) \in \Gamma_1 \times J
\]

where $t$ is time, $c(x, t)$ is solute concentration, $\mathbf{v}(x)$ is a spatially variable steady-state velocity field, $s(x, t)$ is a source-sink term, and $D$ is a constant dispersion tensor assumed to be a function of the mean velocity $\bar{\mathbf{v}}$. $\Gamma_1$ indicates the inflow or noflow boundary where $\mathbf{v} \cdot \mathbf{n} < 0$ and $\bar{c}(x, t)$ is the concentration value specified on this boundary. The transport equation can be written in the Lagrangian form:

\[
\frac{Dc}{Dt} = \nabla \cdot (D \nabla c) + s
\]

where the material derivative $Dc/Dt$ represents the rate of change over time along a particle pathline. The first step of the Eulerian-Lagrangian procedure is an advective step, which derives concentrations at the origins of the $P$ pathlines which end at the $P$ nodes of a fixed computational grid. We trace pathlines to their origins by integrating the velocity backwards over time (from time $t_{n+1}$ to time $t_n$), starting at each node location $x_p$. The velocity integration is carried out with a fourth-order Runge-Kutta algorithm which relies on interpolated velocities evaluated along each pathline. The concentration $c_{pn}$ at the origin of pathline $p$ is obtained by interpolating concentrations at nearby grid nodes which were computed at time $t_n$. 

The second step of the solution procedure is a dispersive step which derives the unknown nodal concentrations at time $t_{n+1}$ implicitly from the following discrete equation:

$$c_{p,n+1} - \Delta t \left[ \nabla \cdot (D \nabla c) + s \right]_{(x_p(t_{n+1}), t_{n+1})} = c_{pn}$$

(3)

where $c_{pn}$ is the known concentration at the origin of pathline $p$ (obtained from the first step of the Eulerian-Lagrangian procedure) and the derivatives in the bracketed term are evaluated at $x_p$ and $t_{n+1}$. We solve this equation with an implicit finite difference procedure which provides values of $c_{p,n+1}$ at time $t_{n+1}$ at the P nodes of the computational grid. This solution can then be used to initialize the next computational step, which moves from $t_{n+1}$ to $t_{n+2}$ [Ruan and McLaughlin, 1997].

Eulerian-Lagrangian solution procedures require both velocity and concentration values at non-nodal locations. These values are typically computed by interpolating known velocities and concentrations available at nearby grid nodes. It is possible to express a general spatial interpolation function in higher dimensional space as a tensor product presentation which has the following form in two dimensions:

$$P(x, y) = \sum_{i=1}^{L_1} \sum_{j=1}^{L_2} \Phi_i(x) \Phi_j(y) a_{ij}$$

(4)

where $\Phi_i(x)$ and $\Phi_j(y)$ are univariate basis functions in the $x$ and $y$ directions. The coefficients $a_{ij}$ are derived from discrete data values at the nodes on a local rectangular grid. For present purposes, this is the finite difference grid described above.

The tensor product decomposition makes it possible to view the multi-dimensional interpolation problem as a sequence of much simpler univariate interpolation problems [Press et al., 1986, Ruan and McLaughlin, 1997]. In order to take full advantage of this decomposition, each univariate interpolator must be expressed in a local (or piecewise) form. This local form is straightforward to define for polynomial (e.g., linear, quadratic, cubic, etc.) interpolators. Press et al. give a local expression for the classical cubic spline interpolator. The local taut spline expression, which is a straightforward extension of the cubic spline, is:

$$P(x) = \alpha z_l + (1 - \alpha) z_{l+1} + C(\Psi, \eta) z''_{l} + D(\Psi, \eta) z''_{l+1}$$

(5)
where \( \alpha = (x_{i+1} - x)/\Delta x \) is a local (non-dimensional) coordinate, 
\( \Delta x = x_{i+1} - x_i \), \( \eta \) is the tension parameter which depends on the
curvature of the solution function [de Boor, 1978], and \( \Psi_l(\alpha; \eta) \) determines the ‘tautness’ of the approximating cubic polynomial [Ruan and McLaughlin, 1997]. The coefficients \( C(\Psi, \eta) \) and \( D(\Psi, \eta) \) are the
same as those of a cubic spline interpolator except when \( \eta \) falls into
a pre-defined range which activates the ‘tautness’ mechanism:

\[
C(\Psi_l, \eta) = \left[ \frac{\Psi_l(\alpha; 1 - \eta) - \alpha}{\Psi''_l(1; 1 - \eta)} \right] \Delta^2 x_l \quad \eta \leq 1/3 \quad (6)
\]

\[
D(\Psi_l, \eta) = \left[ \frac{\Psi_l(1 - \alpha; \eta) - \alpha + 1}{\Psi''_l(1; \eta)} \right] \Delta^2 x_l \quad \eta \geq 2/3 \quad (7)
\]

The Eulerian-Lagrangian algorithm used to generate the taut-spline
results shown in the next section is obtained by combining this local
interpolator with the two-step solution algorithm summarized in (2)
and (3).

**A Two-Dimensional Example**

We use a 2D numerical test problem to compare Eulerian-Lagrangian
solutions computed with linear, quadratic, cubic and taut spline
interpolators. This is a particularly good test because the velocity
is field is heterogeneous, resulting in steep concentration gradients
which are difficult to capture. The heterogeneous velocity field is ob-
tained from a spectrally-based random field generator [Ruan, 1997].
The spatial structure of this field is defined by approximate velocity
(cross)-spectra derived from a linearized analysis of groundwater flow
equation and Darcy’s law. The mean velocity vector has a magnitude
of 0.042 m/day and is aligned with the horizontal direction. The log
hydraulic conductivity is assumed to have a Gaussian spectral den-
sity with a variance is 1.0. The isotropic correlation length is \( 4\Delta x \),
where \( \Delta x \) is the grid spacing. This velocity field conserves mass and
it is practically divergence-free at each nodal location [Ruan, 1997].

In this example we simulate plumes after an instantaneous release
at a specified concentration source. The rectangular computational
domain has a grid size of \( 120\Delta x \times 60\Delta x \), where \( \Delta x = 2.0 \) m is the
Figure 1: Simulated two-dimensional plumes in a spatially variable velocity field. The top row shows contours simulated using linear interpolation (left) and taut spline interpolation (right). The bottom row shows contours simulated using quadratic interpolation (left) and cubic spline interpolation (right). Axes units are in meters.

the grid spacing in both directions. This source is distributed over space according to a Gaussian distribution centered at \((x_0 = 24\Delta x, y_0 = 30\Delta x)\) with a spatial standard deviation of \(1.5\Delta x \times 6\Delta x\) in the horizontal and vertical direction, respectively. The plume is propagated from the left to the right following the mean velocity. The local longitudinal and transverse dispersivities are assumed to be \(\alpha_L = 0.05\) m and \(\alpha_T = 0.05\) m respectively. The simulation period \([0, T]\) is divided into \(N_t = 120\) time steps, where \(\Delta t = 20\) days. This yields an isotropic Peclet number of 40 and a Courant number of 0.42 derived from the mean velocity.
Discussion

Figure 1 shows normalized solute concentration contours obtained from four interpolators at the end of the simulation. The concentrations are normalized by the maximum source concentration $C_{MAX}$. Higher concentrations are indicated with lighter colors. Note that the values associated with the darkest colors are negative (see color scale).

The effects of the heterogeneous velocity field are apparent in all of the plots. The plumes obtained with the cubic and taut spline interpolators have similar peak values, which are significantly higher than the peak obtained from the linear interpolator (upper left). Concentration distributions from the linear interpolator vary more smoothly over space than the other three contours. Negative concentrations are absent in the linear and taut spline cases (upper row) but large patches of negative concentrations arise in quadratic case (lower left). These patches are present both at the upper edge and in the front of the plume. The negative concentrations generated by the cubic spline interpolator also less extensive and are located in different parts of the plume than those generated by the quadratic interpolator. In particular, the cubic spline does much better capturing concentration gradients at the leading edge of the plume.

Previous numerical studies suggest that solution accuracy using linear interpolation may be improved through adaptive grid refinement at locations of sharp concentration gradients [Yeh and Chang, 1992]. But adaptive grid refinement is difficult to implement and computationally expensive to identify such locations. Figure 1 shows higher-order splines avoid the need for grid refinement in this example. The taut spline can further reduce artificial oscillations by detecting the existence of 'extraneous' inflection points (these occur when the parameter $\eta$ lies is outside the range $[1/3, 2/3]$). The classical cubic spline approximation is not able to maintain proper data curvature at such points [de Boor, 1978]. In such cases, the taut spline polynomial provides a smoother and more accurate approximation than the cubic spline.

We have conducted more extensive (and more quantitative tests) which suggest that it is important to choose a good interpolator in the Eulerian-Lagrangian procedure. We argue in Ruan and McLaughlin [1997] that the success of the Eulerian-Lagrangian solution depends primarily on the accuracy of the concentration interpolator. The
cubic spline interpolator can provide better spatial accuracy than the linear and quadratic interpolators, and the taut spline interpolator can further decrease oscillatory negative concentrations at the expense of a small increase in numerical dispersion and computational effort. It is computationally efficient to use cubic and taut splines expressed in a local tensor product form which decomposes the multi-dimensional interpolation problem into a sequence of simpler one-dimensional interpolation problems. We believe the two spline interpolators discussed here are very attractive options for convection-dominated problems, particularly when the underlying velocity field is highly variable.

Acknowledgment

The first author wishes to thank Peter van Bemmel of GeoQuest, Schlumberger for his support of this manuscript.

References