Motion in an Externally Generated Velocity Field

David Frey

26.10.2010

Motivation

Now that we know how to represent a surface with implicit functions new questions arise. Imagine we have a balloon representation and know the speed on the surface during its inflation. How do we deal with such a process on our comuter?

Formulation

Let $V(x) = (u, v, w)^T$ be a velocity field containing the speed of every x on the surface of our object (i.e. $\phi(x) = 0$). One formulation of the problem could be

$$\frac{dx}{dt} = V(x) \tag{1}$$

which is known as the *Lagragian* formulation. In order to use this formula we need to resolve infinitely many points on the surface into a finite set of pieces. These pieces can be lines in two dimensions or triangles in three dimensions.

Now the problem is reduced to moving the endpoints of the pieces along the velocity field V. As long as the connectivity does not change this is no problem. But even simple velocity fields can distort the surface dramatically, in this case the dicretization has to be refreshed periodically in order to deal with these deformations by smoothong the surface where necessary.

Another problem occurs when the topolgy changes during the process, e.g. the object splits up into two objects or several objects merge into one. Such a "surgical operation" can become quite complicated (and are not content of our seminar...). These methods using the Lagrangian formula are know as *front tracking* methods.

Given the complicated problems that may occur using front tracking methods we consider another formulation, which uses the singed distance function ϕ not only to represent the surface, but also to evolve it over time. The so called *Eulerian* formulation (also *Level Set* equation)

$$\phi_t + V \cdot \nabla\phi = 0 \tag{2}$$

can be used to do this. This representation is quite smart, because the conectivity is not important at all and no surgical operations are needed.

But it can be problematic to use this formula, when the velocity field is only defined on the surface (as we assumed). Luckily V is often already defined throughout the domain by the physical laws or the like. In fact we only need a band around the surface to compute the solution properly.

Another difficulty will arise with non continous velocity fields. Consider for example $V = (1, 0, 0)^T$ on the surface and vanishing everywhere else. As $\phi(x_i) \neq 0$ almost everywhere the surface will not move. Notice that this is also the case if V is the same on the surface and continuous on a band of size ϵ and zero elsewhere, as long as Δx is not considerably larger than ϵ. Given this circumstance it would be nice to reduce the variation of the velocity field to a minimum, wouldn't it?

Let us point out some things first: To begin with the values of Von the surface cannot be changed, because they dictate the movement of the surface, regardless of the variation of the velocity field. But if we cannot resolve the variation of the valocity properly the result will not be accurate. Second the velocity off the interface is irrelevant for us, even if it is inherited by some physical calculation, otherwise the Lagangian formula (1) could never solve our problem.

That is to say the velocity variation normal to the surface can be omitted whereas the variaton tangetial to the surface has to be well resolved because it regulates the movement. Thus the minimal variation of V will be achieved by restricting the velocity to be constant in normal direction (which means zero variation). As this generally leads to a multivalued velocity field another approach seems sensible: if we assign $V(x)$ the value of $V(x_c)$ with x_c the closest point on the surface to x the velocity is off the interface is approximately constant.

Spatial Discretization

Upwind Differencing

We have defined ϕ and V on every grid point on the surface we now want to employ numerical methos to calculate the surface evolution. But first some notation: $\phi^n := \phi(t^n)$ contains the values of ϕ at given time t^n, similarly V^n is the speed at time t^n and $t^{n+1} := t^n + \Delta t$. We use the *forward Euler method* as time discretization of (2) until later, which results in

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} + V^n \cdot \nabla \phi^n.$$

At first sight one could try to evaluate the spatial derivates with the finite difference methods mentiond in earlier chapters. Unfortunately this attempt fails, so let us take a look at the expanded formula

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} + u^n \phi^n_x + v^n \phi^n_y + w^n \phi^n_z.$$

Let us go through this dimension by dimension, considering only the $u^n \phi_x^n$ term at a fixed point x_i. If $u_i^n > 0$ the surface moves to the right in x-direction. This suggests that the new value of $\phi(x_i)$ at the end of the time step lies on the left of x_i at the begining of the time step. Likewise the new value of $\phi(x_i)$ will lie to the right of x_i if $u_i^n < 0$. Obvioulsy $D^- \phi$ is the better solutiuon in the first case since $D^+ \phi$ is "looking in the wrong direction" (vice versa in the second case). This method, known as *upwinding,* results in a first order accurate method, because the forward/backward difference are first order, i.e. the errors are $O(\Delta x)$.

Combining forward Euler with upwinding yields a consistent finite difference approximation to the partial differnetial equation in formula (2). The *Lax-Richtmyer equivalence theorem* tells us that such an approximation to a linear partial differnetial equation is convergent if and only if it is *stable.* This can be ensured by the *Courant-Friedrichs-Lewy* condition, which essentially guarantees that the numerical speed $\frac{\Delta x}{\Delta t}$ is at least as fast as the physical speed $|u|$. This yields a time step restriction

$$\Delta t < \frac{\Delta x}{max\{|u|\}}$$

where $max\{|u|\}$ is the maximum over the entire domain. The usual implementation of this condition is by choosing the *CFL* number α with

$$\alpha := \Delta t (\frac{max\{|u|\}}{\Delta x})$$

and $0 < \alpha < 1$ (common choices are 0.9 or 0.5). In multiple dimensions a possible choice for α is

$$\alpha = \Delta t \, max\{\frac{|u|}{\Delta x} + \frac{|v|}{\Delta y} + \frac{|w|}{\Delta z}\} \qquad \text{or} \qquad \alpha = \Delta t \, (\frac{max\{|V|\}}{min\{|u|,|v|,|w|\}}).$$

Hamilton-Jacobi ENO

After developing a first order accurate approximation we want to improve to higher order accuracy. Therefore we will use *essentially nonoscillatory* interpolation of ϕ to gain better approximations to $\phi^-$and ϕ^+. Using *Newton* polynomial interpolation we define the divided differences

$$\begin{array}{ll} D_i^0 \phi := \phi_i & D^1_{i+\frac{1}{2}} \phi := \frac{D^0_{i+1}\phi - D^0_i \phi}{\Delta x} \\ D_i^2 \phi := \frac{D^1_{i+\frac{1}{2}}\phi - D^1_{i-\frac{1}{2}}\phi}{2\Delta x} & D^3_{i+\frac{1}{2}} \phi := \frac{D^2_{i+1}\phi - D^2_i \phi}{3\Delta x}. \end{array}$$

We approximate ϕ and ϕ' with

$$\phi \approx Q_0(x) + Q_1(x) + Q_2(x) + Q_3(x) \qquad \phi' \approx Q_1'(x) + Q_2'(x) + Q_3'(x).$$

In order to find the first order approximation for $(\phi_x^-)_i$ we start with $k = i-1$, for $(\phi_x^+)_i$ with $k = i$. Then the Newton interpolation method delivers

$$Q_1(x) = (D^1_{k+\frac{1}{2}}\phi)(x - x_i) \quad \text{and} \quad Q_1'(x) = D^1_{k+\frac{1}{2}}\phi \tag{3}$$

which is excatly the first order upwinding difference developed earlier.

Next is the second order accurate correction to formulas (3). Here there is a choice to be made: do we include $D_k^2\phi$ or $D_{k+1}^2\phi$ in Q_2? In general interpolation near large variations is dangerous, as it leads to overshoots in the interpolating functions. Hence we compare $|D_k^2\phi|$ and $|D_{k+1}^2\phi|$ and choose the normwise smaller one. So if $|D_k^2\phi| \leq |D_{k+1}^2\phi|$ we set $c = D_k^2\phi$ and $k^* = k-1$ (for later on) and otherwise $c = D_{k+1}^2\phi$ and $k^* = k$. This leads to

$$Q_2(x) = c(x-x_k)(x-x_{k+1}) \quad \text{and} \quad Q_2'(x_i) = c(2(i-k)-1)\Delta x$$

which is a second order accurate approximation.

Similarly the third order approximation is deriverd, comparing $|D^3_{k^*+\frac{1}{2}}\phi|$ with $|D^3_{k^*+\frac{3}{2}}\phi|$ and seting the smaller one as c^*. With this we define

$$Q_3(x) = c^*(x-x_{k^*})(x-x_{k^*+1})(x-x_{k^*+2}) \quad \text{and} \quad Q_3'(x_i) = c^*(3(i-k^*)^2-6(i-k^*)+2)(\Delta x)^2$$

which leads to a third order accurate approximation.

In the process of interpolation we have different possible third order accurate approximations, which can be convex combined cleverly to gain 5th order accuray in smooth regions.

Temporal Descretization

TVD Runge Kutta Scheme

Usually the first order accurate temporal approximation of the forward Euler is sufficient to delivery acceptable result, but in some cases a more accurate method can be helpful. For this reason we will develop a third order accurate *total variation diminishing* Runge Kutta scheme. We assume the forward Euler to be TVD and construct higher order approximations by take additional Eulersteps and convex combining them. The first step goes from t^n to t^{n+1}, the second from t^{n+1} to t^{n+2}.

$$\frac{\phi^{n+1}-\phi^n}{\Delta t} + V^n \cdot \nabla\phi^n = 0 \tag{4}$$

$$\frac{\phi^{n+2}-\phi^{n+1}}{\Delta t} + V^{n+1} \cdot \nabla\phi^{n+1} = 0 \tag{5}$$

After that second order accuracy of the TVD scheme is achieved through averaging the two solutions

$$\phi^{n+1} = \frac{1}{2}\phi^n + \frac{1}{2}\phi^{n+2}$$

For third order accuracy we take the two approximations from (4) and (5) and average differently

$$\phi^{n+\frac{1}{2}} = \frac{3}{4}\phi^n + \frac{1}{4}\phi^{n+2}$$

then caluclate $\phi^{n+\frac{3}{2}}$ from

$$\frac{\phi^{n+\frac{1}{2}} - \phi^{n+\frac{3}{2}}}{\Delta t} + V^{n+\frac{1}{2}} \cdot \nabla \phi^{n+\frac{1}{2}} = 0$$

and average again

$$\phi^{n+1} = \frac{1}{3}\phi^n + \frac{2}{3}\phi^{n+\frac{3}{2}}.$$

Altough there are fourth order accurate schemes in practice there seems to be no significantly improved result in comparison to third order accuracy.