Numerical approximations of derivatives and integralls

Gergely Kristóf

5-th September 2012

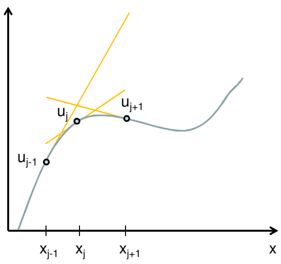
Finite difference method error and convergence

Representative values: x_j, u_j

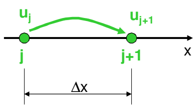
Obtain the change of the solution from the derivatives: $u_{j+1}-u_{j-1}$

The approximation error reduces with reduced interval size.

One scheme is better than the other...



Forward Differencing Scheme (FDS)



From the Taylor polynomial we can express a differencing scheme of first order accuracy:

$$u'_j = \frac{u_{j+1}-u_j}{\Delta x} + o(1)$$

Note that, the error term is one degree of magnitude higher.

Taylor polynomial of the solution from point j to point j+1:

$$u_{j+1} = u_j + u'_j \Delta x + o(\Delta x)$$

This is an integration scheme of first order accuracy.

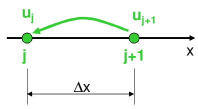
When the differencial equation is given in the explicit form:

$$u'_j = f(u_j, x_j)$$

we can integral step by step, by assuming:

$$u_{j+1} \cong u_j + f(u_j, x_j)\Delta x$$

Backward Differencing Scheme (BDS)



When F is evaluated in j+1, we may end up with a more complicated expression for u_{j+1}. This kind of discretization is called implicit:

Another first order scheme:

$$u_j = u_{j+1} + u'_{j+1}(-\Delta x) + o(\Delta x)$$

from the backward Euler scheme we get:

$$u'_{j+1} = \frac{u_{j+1}-u_j}{\Delta x} + o(1)$$

Now, we assume the differential equation is given in the following form:

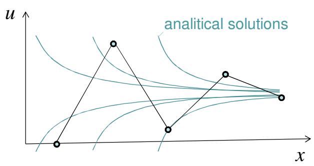
$$F(u', u, x) = 0$$

unknowns

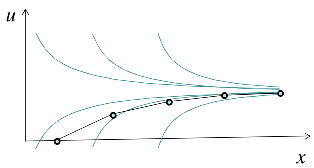
$$F(u'_{j+1}, u_{j+1}, x_{j+1}) = 0 \longrightarrow F\left(\frac{u_{j+1}-u_j}{\Delta x}, u_{j+1}, x_{j+1}\right) \cong 0$$

Different behavior…

Physical processes lead to a temporal equilibrium in many cases.

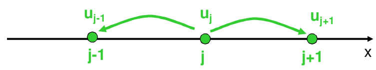


Explicit Euler method:



Implicit Euler method:

Central Differencing Scheme (CDS)



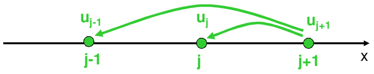
$$u_{j+1} = u_j + u'_j \Delta x + u''_j \frac{\Delta x^2}{2} + o(\Delta x^2)$$

$$u_{j-1} = u_j + u'_j(-\Delta x) + u''_j \frac{\Delta x^2}{2} + o(\Delta x^2)$$

$$u'_j = \frac{u_{j+1}-u_{j-1}}{2\Delta x} + o(\Delta x)$$

Extensively used **in CFD for spatial discretization**.

An implicit differencing scheme with second order accuracy



$$u_j = u_{j+1} + u'_{j+1}(-\Delta x) + u''_{j+1}\frac{\Delta x^2}{2} + o(\Delta x^2)$$

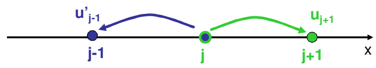
$$u_{j-1} = u_{j+1} + u'_{j+1}(-2\Delta x) + u''_{j+1}\,2\Delta x^2 + o(\Delta x^2)$$

$$u_j - \frac{u_{j-1}}{4} = \frac{3}{4}u_{j+1} + u'_{j+1}\left(-\frac{\Delta x}{2}\right) + o(\Delta x^2)$$

$$u'_{j+1} = \frac{\frac{3}{2}u_{j+1} - 2u_j + \frac{1}{2}u_{j-1}}{\Delta x} + o(\Delta x)$$

Can be used for discretizing the **boundary layer equation**.

Adams-Basforth scheme



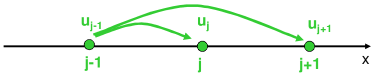
$$u_{j+1} = u_j + u'_j\,\Delta x + u''_j\frac{\Delta x^2}{2} + o(\Delta x^2)$$

$$u'_{j-1} = u'_j + u''_j(-\Delta x) + o(\Delta x) \qquad / \; + \ldots \times \frac{\Delta x}{2}$$

$$u_{j+1} = u_j + \frac{3}{2}u'_j\,\Delta x - \frac{1}{2}u'_{j-1}\,\Delta x + o(\Delta x^2)$$

An explicit integrating scheme with second order accuracy
It is often used **for integrating the Navier-Stoket equations**.

A 2 step 2nd order explicit Runge-Kutta type scheme



1st step: Use the Euler method for geting into point j:

$$u_j = u_{j-1} + u'_{j-1}\,\Delta x + o(\Delta x)$$

Evaluate the derivative in point j:

$$d = f\left(u_j + o(\Delta x), x_j\right) = f\left(u_j, x_j\right) + \left.\frac{df}{du}\right|_{u_j, x_j} \cdot o(\Delta x) = u'_j + o(\Delta x)$$

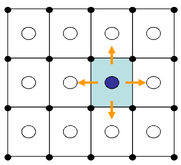
2nd step: Use CDS scheme around point j:

$$u_{j+1} = u_{j-1} + d\,2\,\Delta x + o(\Delta x^2) = u_{j-1} + u'_j\,2\,\Delta x + o(\Delta x^2)$$

Can be used for calculating compressible flows (**eg. Lax-Wendroff method**).

Finite volume method

Discrete solution



U: volume intensity of an arbitrary conserved quantity.

$$\frac{\partial}{\partial t}\int_V U\,dV + \oint_A \vec{F}\cdot d\vec{A} = \int_V S_V\,dV + \oint_A \vec{S}_A \cdot d\vec{A}$$

The conserved quantity per init mass of fluid:

$$\Phi = U/\rho$$

Convective and conductive fluxes:

$$\vec{F}_C = \rho\,\Phi\,\vec{v} \qquad \vec{F}_D = -\Gamma\nabla\Phi$$

$$\frac{\partial}{\partial t}\int_V \rho\,\Phi\,dV + \oint_A \rho\,\Phi\,\vec{v}\cdot d\vec{A} = \oint_A (\Gamma\nabla\Phi + \vec{S}_A)\cdot d\vec{A} + \int_V S_V\,dV$$

Fluxes are **evaluated** on the element faces.
Finite volume method is conservative: discretization errors do not produce or destroy conserved physical properties. Conservation equations are exactly fulfilled on the computational domain.

Discretization of the Navier-Stokes equation is rather difficult on this way…

$$\frac{\partial\rho}{\partial t} + \frac{\partial\rho u}{\partial x} + \frac{\partial\rho v}{\partial y} + \frac{\partial\rho w}{\partial x} = 0$$

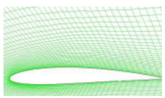
$$\frac{\partial\rho u}{\partial t} + \frac{\partial\rho u^2}{\partial x} + \frac{\partial\rho uv}{\partial y} + \frac{\partial\rho uw}{\partial y} = -\frac{\partial p}{\partial x} + \rho g_x + \frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}\right) + \frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial z}\right)$$

$$\frac{\partial\rho v}{\partial t} + \frac{\partial\rho vu}{\partial x} + \frac{\partial\rho v^2}{\partial y} + \frac{\partial\rho vw}{\partial y} = -\frac{\partial p}{\partial y} + \rho g_y + \frac{\partial}{\partial x}\left(\mu\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial y}\right) + \frac{\partial}{\partial z}\left(\mu\frac{\partial v}{\partial z}\right)$$

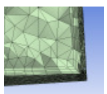
$$\frac{\partial\rho w}{\partial t} + \frac{\partial\rho wu}{\partial x} + \frac{\partial\rho wv}{\partial y} + \frac{\partial\rho w^2}{\partial y} = -\frac{\partial p}{\partial z} + \rho g_z + \frac{\partial}{\partial x}\left(\mu\frac{\partial w}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial w}{\partial y}\right) + \frac{\partial}{\partial z}\left(\mu\frac{\partial w}{\partial z}\right)$$

In some cases more complex meshes are necessary for efficient solution

Curvilinear, stretched



Unstructured, hybrid



Spatial derivatives in finite volume methods

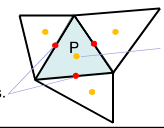
The generic transport equation in integral form:

$$\frac{\partial\rho\phi}{\partial t} + \nabla\cdot(\rho\phi\vec{v}) = \nabla\cdot\vec{S}_A + \nabla\cdot(\Gamma\nabla\phi) + S_v$$

In which Φ is the mass concentration of a conserved quantity (eg. in kg/kg).

Spatial derivatives are always in div(...), grad(...) or div(grad(...)) forms.
We only need to look for the discrete approximations of these operators, which is done - in the case of finite volume method - on the basis of surface and volume integrals along with some spatial interpolations.

The numerical mesh around the cell having its center in point P:



Face centroids.
Defined by surface vectors.

Cell centroid.
Here we store ϕ_P.

Anything can be interpolated from cells to surfaces...

Approximation of the divergence operator

From the volume integral of the divergence operator we can obtain the cell average of the divergence term.
The Gauss-Ostrogradskij theorem for a vector quantity u:

$$\int_V \nabla \cdot \underline{u}\, dV = \oint_A \underline{u} \cdot d\underline{A}$$

For simplicity, we denote components of u vector by u_i. The cell-average of the divergence operator is now:

$$\tilde{\nabla} \cdot u_i = \frac{\sum_k \int_{A_k} u_\perp dA}{V_P}$$

in which A_k are the faces of the cell. The surface integral for each face is a scalar product:

$$\int_{A_k} u_\perp dA = \sum_{i=1}^{3} u_i dA_{k,i}$$

in which u_i is one component of u interpolated to the cell surface.

Gradient

A direct consequence of the Gauss-Ostrogradskij theorem:

$$\int_V \nabla \phi\, dV = \oint_A \phi \cdot d\underline{A}$$

The i-th component of the approximate gradient can be evaluated according to the following expression:

$$\tilde{\nabla}\Big|_i \phi = \frac{\sum_k \int_{A_k} \phi dA_i}{V_P}$$

A_i is the i-th component of the surface vector in Descartes system.

The approximate Laplacian

$$\Delta\phi = \nabla \cdot \nabla \phi$$

When calculating the discrete approximation of the operator the gradient must be interpolated onto the face centroids. This is denoted by < > in the following formula:

$$\tilde{\Delta}\phi = \tilde{\nabla} \cdot \left\langle \tilde{\nabla}\Big|_i \phi \right\rangle$$

For most field variables - excepting for the pressure field – the face normal component of the gradient vector can be calculated on a more simple way: from ϕ values stored in the centers of the adjacent cells.
In this case the discrete form of the Laplacian operator can be calculated as a linear combination of ϕ_P and the neighboring ϕ values:

$$\tilde{\Delta}\phi = A_P\, \phi_P + \sum A_\ell\, \phi_\ell$$

In which A_P are constant values, depending only on the mesh parameters.