A 3D Finite Volume Lagrangian Scheme

Gilles CARRÉ, Stéphane DEL PINO,
Bruno DESPRÉS and Emmanuel LABOURASSE

bruno.despres@cea.fr

CEA
September 2007
Euler equations in Lagrangian coordinates

Lagrangian compressible gas dynamics: \( \tau = \frac{1}{\rho}, \quad p = (\gamma - 1) \frac{\varepsilon}{\rho} \)

\[
\begin{align*}
\rho d_t \tau - \nabla \cdot \mathbf{u} &= 0, \\
\rho d_t \mathbf{u} + \nabla p &= 0, \\
\rho d_t e + \nabla \cdot (p \mathbf{u}) &= 0.
\end{align*}
\]

- Lagrangian fluid dynamics is critical for ICF.
- Finite Volume cell centered schemes have a great potential for ALE developments and remeshing techniques since \( \rho, \mathbf{u} \) and \( e = \varepsilon + \frac{1}{2} |\mathbf{u}|^2 \) are discretized at the same place: the density \( \rho_j^n \), the velocity \( \mathbf{u}_j^n \) and the total energy \( e_j^n \) are the unknowns given in cell \( j \).
- Presentation of ongoing research about the 3D extension.
The Lagrangian mass is \( M_j = s_j(t) \rho_j(t) \). Define \( C_{jr} = l_{jr} \mathbf{n}_{jr} \in \mathbb{R}^2 \). Then

\[
    s_j = \frac{1}{2} \sum_r (C_{jr}, x_r), \quad s'_j(t) = \sum_r (C_{jr}, u_r), \quad \text{and} \quad \sum j C_{jr} = 0.
\]

The idea is to define \( C_{jr} \in \mathbb{R}^3 \) in 3D for general cells (on the left).
One can construct the vectors $C_{jr} \in \mathbb{R}^3$ such that the volume of cell $j$ is

$$V_j = \frac{1}{3} \sum_r (C_{jr}, x_r), \quad V_j'(t) = \sum_r (C_{jr}, u_r), \quad u_r(t) = x'_r(t).$$

For tetrahedra $C_{jr}$ are \textit{uniquely} defined.

For other cell types $C_{jr}$ are \textit{not} uniquely defined.

In any case, one has $\sum_j C_{jr} = 0$. 
Define the mass of the lagrangian cell

\[ M_j = \rho_j(t)V_j(t). \]

One has

\[ M_j \frac{d}{dt} \left( \frac{1}{\rho_j(t)} \right) = V'_j(t) = \sum_r (C_{jr}, u_r). \]

Therefore

\[ \sum_r (C_{jr}, u_r) \approx \int_{\partial j} (u, n) d\sigma \quad \left( = \int_j \nabla \cdot u \, dx \right) \]

is our Finite Volume discretization of the divergence operator.

\[ \implies \text{The } C_{jr} \text{'s give the discretization of the divergence operator.} \]
Discretization of the gradient

We use the idea of a compatible discretization of all discrete operators. One has the formula

$$\sum_r C_{jr} p_r \approx \int_{\partial j} n p \, d\sigma \quad \left(= \int_j \nabla p \, dx \right).$$

We have also

$$\sum_r (C_{jr}, u_r) p_r \approx \int_{\partial j} (u, n) p \, d\sigma = \int_{\partial j} (u, pn) \, d\sigma \quad \left(= \int_j \nabla \cdot (pu) \, dx \right).$$

This gives the structure of the Finite Volume Lagrangian scheme

$$\begin{cases} 
V_j'(t) = \sum_r (C_{jr}, u_r), \\
M_j u_j'(t) = - \sum_r C_{jr} p_{jr}, \\
M_j e_j'(t) = - \sum_r (C_{jr}, u_r) p_{jr}. 
\end{cases}$$
Construction of the fluxes

Riemann-invariant-like formula: \[ p_{jr} - p_j + \rho_j c_j (u_r - u_j, n_{jr}) = 0. \]

Sum of forces is locally zero: \[ \sum_j C_{jr} p_{jr} = \sum_j |C_{jr}| n_{jr} p_{jr} = 0. \]

There is a unique $u_r$ solution magic formulas.
Solution

After elimination of the $p_{jr}$’s, the equation for $u_r$ is

$$
\left( \sum_j \rho_j c_j \left| C_{jr} \right| n_{jr} \otimes n_{jr} \right) u_r
= \left( \sum_j C_{jr} p_j \right) + \left( \sum_j \rho_j c_j \left| C_{jr} \right| n_{jr} \otimes n_{jr} u_j \right).
$$

Since the matrix is symmetric and positive

$$
A_r = \left( \sum_j \rho_j c_j \left| C_{jr} \right| n_{jr} \otimes n_{jr} \right) = A_r^t > 0
$$

there is a unique solution $u_r$. Then we compute the $p_{jr}$’s thanks to

$$
p_{jr} - p_j + \rho_j c_j (u_r - u_j, n_{jr}) = 0.
$$
Scheme in 3D

- Cell quantities \((M_j, V_j, x_j, \rho_j, e_j, u_j \text{ and } p_j)\) are given
- Compute \(C_{jr}\) on the current configuration
- We take \(\Delta t \leq \min_j \left(\frac{\sum_r |C_{jr}|}{3V_j} c_j\right)\).
- \(A_r = \sum_j \rho_j c_j C_{jr} \otimes n_{jr} = A_r^t > 0\).
- \(u_r = A_r^{-1} \sum_j (C_{jr} p_j + \rho_j c_j C_{jr} \otimes n_{jr} u_j)\).
- \(p_{jr} = p_j + \rho_j c_j (u_j - u_r, n_{jr})\)
- \(u_{jr}^{n+1} = u_j^n - \frac{\Delta t}{M_j} \sum_r C_{jr} p_{jr}\)
- \(e_{jr}^{n+1} = e_j^n - \frac{\Delta t}{M_j} \sum_r (C_{jr}, u_r) p_{jr}\)
- \(x_r^{n+1} = x_r^n + \Delta t u_r\)
- Update \(V_j, \rho_j \text{ and } p_j\)
Conservativity

- Mass conservation: the scheme imposes $M_j$ to be constant.
- Momentum conservation:

\[
\sum_j M_j u_j^{n+1} = \sum_j M_j u_j^n - \sum_j \Delta t \sum_r C_{jr} p_r \\
= \sum_j M_j u_j^n - \sum_j \Delta t \sum_r \sum_j C_{jr} p_r \\
= \sum_j M_j u_j^n
\]

- Total energy conservation:

\[
\sum_j M_j e_j^{n+1} = \sum_j M_j e_j^n - \sum_j \Delta t \sum_r (C_{jr}, u_r) p_r \\
= \sum_j M_j e_j^n - \sum_j \Delta t \sum_r \left( \sum_j C_{jr} p_r, u_r \right) \\
= \sum_j M_j e_j^n
\]

\[\Rightarrow\] the scheme is **locally** conservative.

It is also stable and consistent.
Second order extension

- **Goals:** Improve the accuracy of the scheme and diminish scheme dissipation.

- **Method:** Muscl reconstruction + Van-Leer slope limiter:

\[
\tilde{u}_j(x) = u_j + \Phi^{vl} \nabla u_j (x - x_j), \quad \tilde{p}_j(x) = p_j + \Phi^{vl} \nabla p_j \cdot (x - x_j)
\]

\(\Phi^{vl} \in [0, 1]\) is chosen so that \(\tilde{p}_j\) and \(|\tilde{u}_j|\) respect some maximum principle \(\text{à la Ducowicz}\).

The scheme is order 1 in time, so we need smaller time steps to avoid the anti-dissipative term of the equivalent equation.
Second order extension

- Second Method. Muscl reconstruction + LW procedure + Van-Leer slope limiter (Dukovicz):
  - \( \bar{u}_j(x) = u_j + \Phi^v(1 - \nu) \nabla u_j(x - x_j) \)
  - \( \bar{p}_j(x) = p_j + \Phi^v(1 - \nu) \nabla p_j \cdot (x - x_j) \)

Second order fluxes are still

- \( u_r = A_r^{-1} \sum_j (C_{jr} \bar{p}_j(x_r) + \rho_j c_j C_{jr} \otimes n_{jr} \bar{u}_j(x_r)) \).

- \( p_{jr} = \bar{p}_j(x_r) + \rho_j c_j (\bar{u}_j(x_r) - u_r, n_{jr}) \)

The scheme is formally order 2 in time and space for simple 1D problems.
Numerical results

We compute the results of some basic shock tube problems in various geometries

Sod shock tube

\[ \gamma = 1.4, u = 0 \]
\[ \begin{cases} p(x) = 1 & \rho(x) = 1 & \text{if } x < 0.5, \\ p(x) = 0.1 & \rho(x) = 0.125 & \text{else}. \end{cases} \]

Shestakov shock tube

\[ \gamma = \frac{5}{3} \quad u = 0 \quad p = \frac{2}{3} \times 10^{14} \quad x_{\text{min}}(0) = 0 \quad x_{\text{max}}(0) = 0.75 \]
\[ u(x_{\text{max}}(t)) = 2.5 \times 10^7 \]
\[ u(x_{\text{min}}(t)) = \begin{cases} 10^8 \left( (1 - 10^8 t)^{-0.25} - 1 \right), & \text{if } 0 \leq t < 0.9 \times 10^{-8}, \\ 10^8 (10^{0.25} - 1) \approx 7.8 \times 10^7, & \text{if } 0.9 \times 10^{-8} \leq t < 10^{-8}, \\ 0, & \text{else}. \end{cases} \]

Sod shock tube in 2D and 3D

\[ \gamma = 1.4, u = 0 \]
\[ \begin{cases} p(r) = 1 & \rho(r) = 1 & \text{if } x > 0.5, \\ p(r) = 0.1 & \rho(r) = 0.125 & \text{else}. \end{cases} \]
Sod shock tube - 1D

Density 100 cells, $t = 0.2\text{s}$
Shestakov shock tube - 1D

Density 100 cells, \( t = 10.4 \times 10^{-9} s \)
Sod shock tube - 2D

Density $100 \times 50$ cells, $t = 0.2s$
Density. Order one 187 500. **Order two 21 600 cells.** $t = 0.2s$. Parallel computation
Hourglass with Noh

This has already been identified in 2D.

Noh problem in 2D: \( \gamma = \frac{5}{3}, \rho = 1, p = 0(\approx 10^{-6}), \mathbf{u} = -\mathbf{e}_r. \)

Using a polar mesh, one obtains
A fix-up

- $c_j \ll 1 \implies A_r$ nearly singular.

- Noh: $c_j \approx 10^{-3} \implies A_r = \sum_j \rho_j c_j C_{jr} \otimes n_{jr} \approx 0.$

$$u_r = A_r^{-1} \sum_j \left( C_{jr} p_j(x_r) + \rho_j c_j C_{jr} \otimes n_{jr} \bar{u}_j(x_r) \right).$$

- Idea: increase $c$ when it is too small for the fluxes calculus (more dissipative scheme).
In order to increase the time step, we can replace

\[ p_{jr} - p_j + \rho_j c_j (u_r - u_j, n_{jr}) = 0 \]

by

\[ p_{jr} - p_j + \lambda_j \rho_j c_j (u_r - u_j, n_{jr}) = 0. \]

It can be shown that the optimal \( \lambda_j \) is

\[ \lambda_j = \sum_r \frac{|C_{jr}|}{\Lambda} \in [1, \sqrt{3}], \]

where \( \Lambda \) is the maximum eigenvalue of

\[ \sum_r \frac{C_{jr} \otimes C_{jr}}{|C_{jr}|}. \]

Finally \( \Delta t \leq \min_j \left( \lambda_j \frac{\sum_r |C_{jr}|}{3V_j} c_j \right) \).
Reduce connectivity points in 3D

- The low-Mach test case is similar to the 2D one published in the thesis of Raphael Loubere

\[ |u| \ll c. \]

It consists of a 3D shell implosion, computed with a perfect gas law, and with a prescribed pressure on the internal and external faces of the shell. Since the computation is done with \( \frac{1}{8} \) of the shell, sliding walls are used on other boundaries.

- The reduced connectivity point is important.
Reduce connectivity points in 3D
Reduce connectivity points in 3D
Reduce connectivity points in 3D

\( t \mapsto d_r(t) \) is the sphericity error compute on the internal boundary.

AMR (and ALE) techniques are easy to use for cell-centered schemes.
For this problem it enhances the final quality and exerces some control on the reduce connectivity point.
Conclusions et perspectives

- The whole scheme relies on the definition of $C_{jr}$.
  - This is straightforward only for tetrahedrons.
  - No unique definition of $C_{jr}$ for other meshes in 3D.
  - So we need to compare all possible definition of theses quantities.

- Boundary conditions and second order Muscl reconstruction completed.

- We begin the evaluation of the coupling with ALE and remeshing.

- The reduce connectivity point (in 3D) behaves well in conjunction with AMR.

- We shall start coupling with other physics (diffusion) to test this scheme for ICF simulations.

- ...