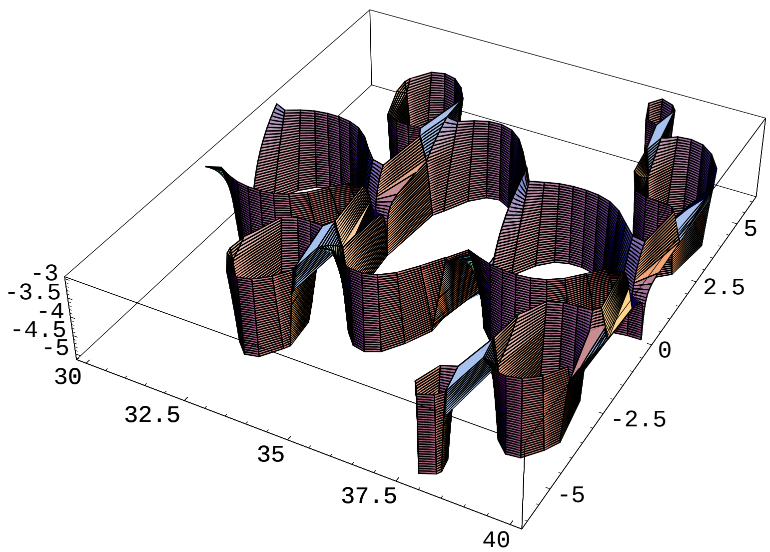
Half Explicit Methods: the Next Generation of Low MACH Number Time-Integrators ?



P.P. PÉBAY[*], J. POUSIN[‡] & H.N. NAJM[*]
[*]Sandia National Laboratories, Livermore CA, U.S.A.
[‡]Département de Mathématiques, I.N.S.A., Lyon, France

This work is mostly supported by the U.S. Department of Energy, Office of Defence Programs & Office of Basic Sciences

Outline

- The low MACH number approximation
- The geometric point of view
- Pros and cons of the geometric point of view
- Space-discretization
- Low MACH flows as a DAE2 system
- Half-Explicit Methods
- HEM: where is the implicitness?
- An equivalent formulation of the scheme
- Implicit solves in the low MACH case
- Numerical perspectives

The low MACH number approximation

The low MACH number approximation assumes that the pressure of the flow p can be written as:

$$p(x,t) = P_0(t) + p_1(x,t)$$

with $\frac{p_1}{P_0} = \mathrm{O}(M^2)$ where M is the MACH number. p_0 is spatially constant, and also constant in time if the domain is open. This approximation is acceptable if M is sufficiently small, say $M < 0.3$.

Ignoring body forces, the governing equations for non-reacting low MACH flows are:

$$\left\{\begin{array}{rcl} \partial_t(\rho u) & = & \mathrm{div}\tau - \mathrm{div}(\rho u \otimes u) - \mathrm{grad}\, p_1 \\ \partial_t T & = & -\langle u, \mathrm{grad} T\rangle + \dfrac{\mathrm{div}(\alpha\, \mathrm{grad} T)}{\rho c_p} \\ \partial_t \rho & = & -\langle u, \mathrm{grad}\rho\rangle - \rho\, \mathrm{div} u \\ \rho & = & \dfrac{P_0}{RT} \end{array}\right.$$

The Geometric Point of View

The classical approach for numerically solving the system consists in differentiating the log of the state law with the material derivative operator, from where a temperature derivative arises, and then to use the continuity, leading to

$$\begin{cases} \partial_t(\rho u) &= \operatorname{div}\tau - \operatorname{div}(\rho u \otimes u) - \operatorname{grad} p_1 \\ D_t T &= \dfrac{\operatorname{div}(\alpha \operatorname{grad} T)}{\rho c_p} \\ D_t \rho &= -\rho \operatorname{div} u \\ \operatorname{div} u &= \dfrac{D_t T}{T} \end{cases}$$

whence

$$\begin{cases} \partial_t(\rho u) &= \operatorname{div}\tau - \operatorname{div}(\rho u \otimes u) - \operatorname{grad} p_1 \\ \partial_t T &= -\langle u, \operatorname{grad} T\rangle + \dfrac{\operatorname{div}(\alpha \operatorname{grad} T)}{\rho c_p} \\ \partial_t \rho &= -\langle u, \operatorname{grad} \rho\rangle - \rho \operatorname{div} u \\ \operatorname{div} u &= \dfrac{\operatorname{div}(\alpha \operatorname{grad} T)}{T \rho c_p} \end{cases}$$

Pros and Cons of the Geometric Point of View

The latter formulation is weaker than the initial one, *i.e.*, the constraint in $\mathrm{div}\,u$ is intrisically weaker, results in a larger constraint submanifold, than that in ρ.

The main advantage of this approach is that one has now an explicit expression for $\mathrm{div}\,u$, which is needed anyway for the numerical schemes solving for p_1.

Practically, equations are decoupled and the methods aim at solving hydrodynamic and mass conservation equations for a given temperature while preserving, or attempting to do so, the geometric invariant constituted by the constraint on $\mathrm{div}\,u$. This is generally done by projection methods, implicitly assuming that the state law is therefore automatically enforced.

It has been reported that, in the regions where the velocity is supposed to be divergence-free, such schemes might fail at satisfying this property. In fact, the geometric point of view leads to projecting onto a geometric manifold, larger than the one induced by the initial governing equations.

Space-Discretization

Formally, a space-discretization leads to the following:

$$(\forall k \in K) \quad \left\{ \begin{array}{rcl} \partial_t(\rho^{(k)} u^{(k)}) & = & A((\rho^{(k)})_{k\in K}, (u^{(k)})_{k\in K}) - B((p_1^{(k)})_{k\in K}) \\ \partial_t T^{(k)} & = & C((\rho^{(k)})_{k\in K}, (u^{(k)})_{k\in K}, (T^{(k)})_{k\in K}) \\ \partial_t \rho^{(k)} & = & D((\rho^{(k)})_{k\in K}, (u^{(k)})_{k\in K}) \\ 0 & = & R\rho^{(k)} T^{(k)} - P_0 \end{array} \right.$$

where A, B, C and D are polynomial functions depending on the space-discretization scheme. Defining $y = (y^{(k)})_{k\in K} = ((\rho^{(k)})_{k\in K}, (T^{(k)})_{k\in K}, (u^{(k)})_{k\in K})$, this yields:

$$(\forall k \in K) \quad \left\{ \begin{array}{rcl} \partial_t \left(y^{(k)}\right) & = & f\left(y, (p_1^{(k)})_{k\in K}\right) \\ 0 & = & g\left(y^{(k)}\right) \end{array} \right.$$

or, with $z = (p_1^{(k)})_{k\in K}$,

$$\left\{ \begin{array}{rcl} \partial_t y & = & f(y,z) \\ 0 & = & g(y) \end{array} \right.$$

Low Mach Flows as a DAE2 System

More formally, the problem can be re-written as follows: given $(n,m) \in \mathbb{N}^{*2}$, $f \in \mathscr{C}^0(\mathbb{R}^{n+m}, \mathbb{R}^n)$ and $g \in \mathscr{C}^1(\mathbb{R}^n, \mathbb{R})$), solve the following system of equations with unknows $y \in \mathscr{C}^1(\mathbb{R}_+, \mathbb{R}^n)$ and $z \in \mathscr{C}^0(\mathbb{R}_+, \mathbb{R}^m)$:

$$(\forall t \in \mathbb{R}_+) \quad \begin{cases} y'(t) & = & f(y(t), z(t)) & \qquad y(0) = y_0 \\ 0 & = & g(y(t)) & \qquad z(0) = z_0 \end{cases}$$

cf. E. Hairer & G. Wanner, Solving O.D.E. II, Springer-Verlag (1996)

Our approach consists in trying to solve the "real" DAE2 system rather than a differentiated version of it.

Remark Given such a DAE2, the following equation:

$$\frac{\mathrm{d}g}{\mathrm{d}y}(y) \cdot f(y,z) = g_y(y) \cdot f(y,z) = 0,$$

which arises when combining the differential equation with the time-derivative of the algebraic constraint is called the *hidden constraint*.

Half-Explicit Methods

Half Explicit Methods (HEM) are one-step multi-stage DAE2 time-integrators. Each step with length h of a s-stage scheme is defined by:

$$\begin{cases} (\forall i \in \{1,\dots,s\}) & Y_i = y_0 + h \sum_{j=1}^{i-1} a_{ij} f(Y_j, Z_j) \\ (\forall i \in \{1,\dots,s\}) & 0 = g(Y_i) \\ & y_1 = y_0 + h \sum_{j=1}^{s} b_j f(Y_j, Z_j) \\ & 0 = g(y_1) \end{cases}$$

cf. E. HAIRER, C. LUBICH & M. ROCHE, The Numerical Solution of Differential-Algebraic Systems by RUNGE-KUTTA Methods, Lecture Notes in Mathematics **1409**, Springer-Verlag (1989)

HEM: Where is the Implicitness?

Contrary to appearences, the implicitness in HEM does not fall on the differential but on the algebraic variable:

$$\begin{cases} (\forall i \in \{1,\ldots,s\}) & Y_i = y_0 + h \sum_{j=1}^{i-1} a_{ij} f(Y_j, Z_j) \\ (\forall i \in \{1,\ldots,s\}) & 0 = g(Y_i) \\ & y_1 = y_0 + h \sum_{j=1}^{s} b_j f(Y_j, Z_j) \\ & 0 = g(y_1) \end{cases}$$

Stage 1: y_0, thus Y_1 are known. No implicit solve. Y_1, thus y_0, must be consistent.
Stage 2: y_0 and Y_1 are known. No implicit solve. Z_1 is needed to compute Y_2, whence an implicit solve in Z_1. Then Y_2 is explicit.
... and so on ...
To summarize: each Z_i is needed to compute Y_{i+1}, and finally, Z_s is needed to compute y_1.

An Equivalent Formulation of the Scheme

We re-formulate the scheme in such a way that the "real" implicitness appears, making the implementation more straightforward: assuming $g(y_0) = 0$ and $Y_1 = y_0$,

$$\left\{ \begin{array}{llcl} (\forall i \in \{1,..,s-1\}) & X_i & = & y_0 + h \sum_{j=1}^{i-1} a_{i+1,j} f(Y_j, Z_j) \\ & 0 & = & g(X_i + h a_{i+1,i} f(Y_i, Z_i)) \\ (\forall i \in \{1,..,s-1\}) & Y_{i+1} & = & X_i + h a_{i+1,i} f(Y_i, Z_i) \\ & X_s & = & y_0 + h \sum_{j=1}^{s-1} b_j f(Y_j, Z_j) \\ & 0 & = & g(X_s + h b_s f(Y_s, Z_s)) \\ & y_1 & = & X_s + h b_s f(Y_s, Z_s) \end{array} \right.$$

is equivalent to the "standard" HEM formulation.

Implicit Solves in the Low MACH Case

At each stage i, the following equation must be solved for Z_i:

$$g(X_i + ha_{i+1,i} f(Y_i, Z_i)) = \phi_i(Z_i) = 0$$

which in the low MACH case becomes

$$(\forall k \in K) \quad \phi_i\left(p_{1_i}^{(k)}\right) = 0$$

with

$$f: \begin{array}{ccc} \mathbb{R}^{n+m} & \longrightarrow & \mathbb{R}^n \\ \begin{pmatrix} (\rho^{(k)})_{k\in K} \\ (T^{(k)})_{k\in K} \\ (u^{(k)})_{k\in K} \\ (p_1^{(k)})_{k\in K} \end{pmatrix} & \longmapsto & \begin{pmatrix} A((\rho^{(k)})_{k\in K}, (u^{(k)})_{k\in K}) - B((p_1^{(k)})_{k\in K}) \\ C((\rho^{(k)})_{k\in K}, (u^{(k)})_{k\in K}, (T^{(k)})_{k\in K}) \\ D((\rho^{(k)})_{k\in K}, (u^{(k)})_{k\in K}) \end{pmatrix} \end{array}$$

This results in a non-linear equation for the pressure at each node, which therefore yields to reasonably-sized non-linear systems.

Numerical Perspectives

We have performed numerical tests with the HEM5 scheme (5-stage, 4^{th} order in y, *cf*. V. BRASEY, Half explicit methods for semi-explicit differential-algebraic equations of index 2, PhD Thesis 2664, Université de Genève, CH (1994)).

In particular, with the Modified KAPS Problem (MK), the method has demonstrated its robustness and cost-efficiency. Starting with consistent initial conditions, no drift-off is observed, contrary to what is observed with a scheme based on index-reduction (local state space form ERK4).

We therefore have reasonable hopes that:

1. using the constrained transport formulation for low MACH number flows, and

2. solving it directly with a DAE2 scheme such as a HEM method

will significantly improve the direct numerical simulation of low MACH number flows.