



A totally Eulerian Finite Volume solver for multi-material fluid flows: Enhanced Natural Interface Positioning (ENIP)

Raphaël Loubère, Jean-Philippe Braeunig, Jean-Michel Ghidaglia

► To cite this version:

Raphaël Loubère, Jean-Philippe Braeunig, Jean-Michel Ghidaglia. A totally Eulerian Finite Volume solver for multi-material fluid flows: Enhanced Natural Interface Positioning (ENIP). 2010. hal-00527765v1

HAL Id: hal-00527765

<https://hal.science/hal-00527765v1>

Preprint submitted on 20 Oct 2010 (v1), last revised 21 Oct 2010 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A totally Eulerian Finite Volume solver for multi-material fluid flows : Enhanced Natural Interface Positioning (ENIP)

Raphaël Loubère^{a,*}, Jean-Philippe Braeunig^b, Jean-Michel Ghidaglia^c

^a*CNRS et Université de Toulouse IMT (Institut de Mathématiques de Toulouse), 31062 Toulouse, France*

^b*INRIA Nancy Grand-Est, Equipe CALVI, 615 rue du Jardin Botanique 54600 Villers-lès-Nancy, France*

CEA DIF Bruyères-le Châtel, 91297 Arpajon, France

^c*CMLA, CNRS et ENS de Cachan 61 Av. du Président Wilson Cachan Cedex 94235, France*

Abstract

This work concerns the simulation of compressible multi-material fluid flows and follows the method FVCF-NIP described in the former paper [5]. This Cell-centered Finite Volume method is totally Eulerian since the mesh is not moving and a sharp interface, separating two materials, evolves through the grid. A sliding boundary condition is enforced at the interface and mass, momentum and total energy are conserved. Although this former method performs well on 1D test cases, the interface reconstruction suffers of poor accuracy in conserving shapes for instance in linear advection. This situation leads to spurious instabilities of the interface. The method Enhanced-NIP presented in the present paper cures an inconsistency in the former NIP

*Corresponding author

Email addresses: `raphael.loubere@math.univ-toulouse.fr` (Raphaël Loubère), `braeunig@math.u-strasbg.fr` (Jean-Philippe Braeunig), `jmg@cmla.ens-cachan.fr` (Jean-Michel Ghidaglia)

method that improves strikingly the results. It takes advantage of a more consistent description of the interface in the numerical scheme. Results for linear advection and compressible Euler equations for inviscid fluids are presented to assess the benefits of this new method.

Keywords: Multi-material fluid flow, Finite Volume, Natural Interface Positioning

2010 MSC: 65M08, 76M12, 76N99

1	Contents	
2	1 Introduction	3
3	2 FVCF-ENIP: Finite Volume Characteristics Flux with En-	
4	hanced Natural Interface Positioning technique	6
5	2.1 Governing equations	6
6	2.2 FVCF: Single material scheme	7
7	2.3 FVCF-NIP: Multi-material scheme	8
8	2.4 FVCF-ENIP	14
9	2.4.1 Lagrangian <i>Condensate evolution</i> step	15
10	2.4.2 <i>Reconstruction</i> step	18
11	2.4.3 <i>Projection</i> step	19
12	3 Numerical results	20
13	3.1 Advection context	20
14	3.2 Hydrodynamics context	23
15	4 Conclusion and perspectives	25

16 1. Introduction

17 The two-material compressible hydrodynamics equations (Euler equa-
18 tions) are considered in this work. The flow regime is such that molecular
19 viscosity within materials is neglected: materials are supposed immiscible
20 and separated by sharp interfaces, with perfect sliding between materials.
21 Each material is characterized by its own equation of state (EOS).

22 The formalism of finite volume methods is close to the mechanical view-
23 point, and generic for different types of physical models. Thus, it might be
24 easier to add such models; surface tension or turbulent diffusion for instance.
25 The discretization order is limited, but this method is accurate to simulate
26 hydrodynamic shock waves, because of the consistency between numerical
27 treatment and mechanics.

28 The extension of Eulerian schemes to multi-material fluid flows can be ob-
29 tained by various techniques. One is to introduce the cell mass fraction c_α of
30 material α and let it evolve according to material velocity. The cell is called
31 pure if a material α satisfies $c_\alpha = 1$ and is called mixed if $c_\alpha \in]0, 1[$. Pure
32 cells filled by material α are calculated in the same manner as for the single
33 material method. Mixed cell evolution is computed using a mixing equation
34 of state that takes into account material mass fractions, see *e.g.* [1]. One
35 drawback of this approach is the numerical diffusion of the interface that ob-
36 viates sharp interface capturing. It turns out that for some applications, this
37 drawback is not acceptable since the diffusion of one material into another
38 one will correspond to a different physics. For example the two material
39 could react when a molecular mixture is formed. Hence such a diffusion
40 should occur only for physical reasons and not for numerical ones.

41 In the case of sharp interface capturing methods, the interface is approx-
 42 imated in a mixed cell by a segment by most authors. However more com-
 43 plex curves than straight line or more complex theory (see [6] for instance)
 44 might be used. A famous method using sharp interface reconstruction is
 45 the Lagrange+Remap Finite Volume scheme, initiated in [9] and further im-
 46 proved in [10]. It belongs to the family of so called Volume of Fluid (VOF)
 47 methods. The first step of this method is a Lagrangian scheme, resulting
 48 in a mesh displacement with material velocity. The second step is a multi-
 49 material remapping of Lagrangian mesh onto the original Eulerian mesh, by
 50 exchanging volume fluxes between cells related to the Lagrangian motion of
 51 cell edges. The new interface position in mixed cells is determined using
 52 the partial volumes of the materials and the interface normal vector. The
 53 later is calculated using volume fractions from neighboring cells. Thus the
 54 ratio of each material in volume fluxes is deduced from the multi-material
 55 remapping. Some methods with the same kind of operator splitting are used
 56 for incompressible multi-material fluid flows as in [8]. These methods pro-
 57 vide sharp interface between materials and discontinuous quantities in mixed
 58 cells, allowing large deformations and transient flows. In this context, the
 59 drawback of these Lagrange+Remap methods is the limited accuracy of the
 60 underlying single phase scheme due to diffusion induced by the remapping
 61 step. Moreover, more complex physics at material interfaces such as sliding
 62 effects, is not possible.

63 The FVCF scheme (Finite Volume with Characteristic Flux) has been
 64 introduced in [7] for simulating single phase compressible flows or multi-
 65 phase models without sharp interface capturing. The method described in

[5], so called NIP method (Natural Interface Positioning), is an add-on to the FVCF method in order to deal with multi-material fluid flows with sharp interface capturing. It is a cell centered totally Eulerian scheme, in which material interfaces are represented by a discontinuous piecewise linear curve. A treatment for interface evolution is proposed on Cartesian structured meshes which is locally conservative in mass, momentum and total energy and allow the materials to slide on each others. Discrete conservation laws are written on partial volumes as well as on pure cells, considering the interface in the cell as a moving boundary without any diffusion between materials. A specific data structure called *condensate* is introduced in order to write a finite volume scheme even when the considered volume is made of moving boundaries, i.e. interfaces. This treatment includes an explicit computation of pressure and velocity at interfaces.

In [5] are shown 2D results illustrating the capability of the method to deal with perfect sliding, high pressure ratios and high density ratios. This former method however produces non satisfactory results in the context of advection of geometrical shapes especially when dealing with low Mach numbers. It is however a classical misbehavior of most of advection and reconstruction methods which have a tendency to destroy the shape of advected objects due to numerical approximations. However, this former method gives very poor results when advecting geometrical shapes especially when dealing with low Mach number flows. In this work we propose a new method called ENIP (Enhanced NIP) that is an improvement of the NIP method by a more accurate treatment of condensates. On a very simple example: the advection of a square, an inconsistency in the NIP interface reconstruction method will

91 be exhibited. We will then introduce ENIP that cures this situation. Nu-
 92 merical examples are presented in the last Section to assess the validity and
 93 efficiency of this new approach.

94 **2. FVCF-ENIP: Finite Volume Characteristics Flux with Enhanced** 95 **Natural Interface Positioning technique**

96 *2.1. Governing equations*

The model addressed in this work is the compressible Euler equations in space dimension d that can be written in a conservative form as follows:

$$\frac{\partial}{\partial t}(\rho) + \operatorname{div}(\rho u) = 0, \quad (1)$$

$$\frac{\partial}{\partial t}(\rho u) + \operatorname{div}(\rho u \otimes u + pI) = 0, \quad (2)$$

$$\frac{\partial}{\partial t}(\rho E) + \operatorname{div}((\rho E + p)u) = 0, \quad (3)$$

where ρ denotes the density, $u \in \mathbb{R}^d$ the velocity field, p the pressure, $E = e + |u|^2/2$ the specific total energy and e the specific internal energy. An equation of state of the form $EOS(\rho, e, p) = 0$ or $p = p(\rho, e)$ is provided in order to close the system.

Let us consider a generic conservative form with $\mathbf{V} = (\rho, \rho u, \rho E)^t$ the unknown vector of conservative variables and flux \mathbf{F} is a matrix valued function defined as:

$$\begin{aligned} \mathbf{F} : \mathbb{R}^{d+2} &\longrightarrow \mathbb{R}^{d+2} \times \mathbb{R}^d \\ \mathbf{V} &\longmapsto \mathbf{F}(\mathbf{V}), \end{aligned} \quad (4)$$

for all direction $n \in \mathbb{R}^d$, $\mathbf{F}(\mathbf{V}) \cdot n$ is given in terms of \mathbf{V} by:

$$\mathbf{F}(\mathbf{V}) \cdot n = (\rho(u \cdot n), \rho u(u \cdot n) + pn, (\rho E + p)(u \cdot n)). \quad (5)$$

The compressible Euler equations (1-3) can then be rewritten as:

$$\partial_t \mathbf{V} + \text{div } \mathbf{F}(\mathbf{V}) = 0. \quad (6)$$

97 2.2. FVCF: Single material scheme

98 FVCF method uses a directional splitting on Cartesian structured meshes.
 99 The method is thus detailed for only one generic direction denoted by x . In
 100 d dimensions of space, the algorithm described for direction x has to be
 101 replicated d times, one for each direction. However, this directional splitting
 102 does not modify at all the underlying single material scheme FVCF for pure
 103 cells. In $2D$:

- 104 - variables at $t^{n,x}$ are calculated from those at t^n by the x direction step,
- 105 - variables at t^{n+1} are calculated from those at $t^{n,x}$ by the y direction
- 106 step.

$$\text{Vol}_i \frac{\mathbf{V}_i^{n,x} - \mathbf{V}_i^n}{\Delta t} + A_x (\phi_\ell^n + \phi_r^n) = 0, \quad (7)$$

$$\text{Vol}_i \frac{\mathbf{V}_i^{n+1} - \mathbf{V}_i^{n,x}}{\Delta t} + A_y (\phi_d^n + \phi_u^n) = 0, \quad (8)$$

where the cell volume is Vol_i , the cell face area are A_x and A_y respectively
 normal to x and y directions, up, down, right and left direction fluxes ϕ_u^n ,
 ϕ_d^n , ϕ_r^n , ϕ_ℓ^n calculated with respect of the outgoing normal direction n_d of
 cell face Γ_d in direction d using variables at time t^n , i.e.

$$\phi_d^n = \frac{1}{A_d} \int_{\Gamma_d} \mathbf{F}(\mathbf{V}^n) \cdot n_d dS. \quad (9)$$

107 This flux is further approximated using the finite volume scheme FVCF de-
 108 scribed in [7].

109 *2.3. FVCF-NIP: Multi-material scheme*

One considers multi-material flows. The subcell model addressed here for the multi-material representation is a cell C of volume Vol_C containing n_m different materials, each of them filling a partial volume Vol_C^k such that

$$\sum_{k=1}^{n_m} \text{Vol}_C^k = \text{Vol}_C. \quad (10)$$

110 Cell C is referred to as pure if $n_m = 1$, and as mixed if $n_m > 1$. The interfaces
111 in mixed cells are approximated by segments separating materials into two
112 partial volumes which are pure on both sides of the interface.

113 A partial volume cell-centered variable vector $\mathbf{V}_k = (\rho_k, \rho_k u_k, \rho_k E_k)^t$ and an
114 equation of state $EOS_k(\rho_k, e_k, p_k) = 0$ are also associated with each material
115 labeled by $k \leq n_m$ in the mixed cell.

116 FVCF-NIP method uses a directional splitting scheme for the interface evo-
117 lution without loosing the accuracy of the Eulerian scheme in the bulk of
118 materials. Consequently this scheme is restricted to structured Cartesian
119 mesh.

The multi-material extension proposed in [5] considers the finite volume scheme (7-8) on each partial volume in a mixed cell. The obtained scheme is conservative by construction and is constrained with the same CFL condition as the single material scheme¹. NIP method consists in removing cell edges when this cell contains an interface. Therefore each partial volume is merged with the neighbor pure cells filled with the same material, see Figure 1. Variables in these enlarged partial volumes are obtained by writing the

¹Without such a special treatment the time step would be constrained by the smallest partial volume, which is arbitrarily small.

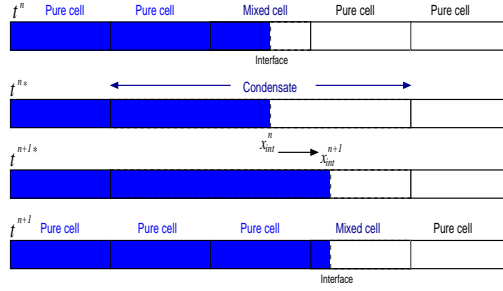


Figure 1: Sketch of a condensate. Evolution of an interface through a cell edge during one time step. Mixed and pure neighbor cells are merged to obtain the so called condensate at fictitious time t^{n*} . Interface evolution is performed within this condensate from t^{n*} to t^{n+1*} . This condensate is then split back into Eulerian cells.

conservation laws on the merged volumes

$$\overline{\text{Vol}}_1 = \text{Vol}_1 + \text{Vol}_{\text{pure } 1}, \quad (11)$$

$$\overline{\text{Vol}}_2 = \text{Vol}_2 + \text{Vol}_{\text{pure } 2}, \quad (12)$$

then on the conserved variables

$$\overline{\mathbf{V}}_1 = \frac{\text{Vol}_1 \mathbf{V}_1 + \text{Vol}_{\text{pure } 1} \mathbf{V}_{\text{pure } 1}}{\overline{\text{Vol}}_1} \quad (13)$$

$$\overline{\mathbf{V}}_2 = \frac{\text{Vol}_2 \mathbf{V}_2 + \text{Vol}_{\text{pure } 2} \mathbf{V}_{\text{pure } 2}}{\overline{\text{Vol}}_2}. \quad (14)$$

120 This set of cells is associated with its left and right single material fluxes ϕ_ℓ
 121 and ϕ_r . Internal cell edges are forgotten, considering only enlarged volumes
 122 $\overline{\text{Vol}}_1$ and $\overline{\text{Vol}}_2$ and averaged variables $\overline{\mathbf{V}}_1$ and $\overline{\mathbf{V}}_2$, separated by an interface;
 123 this system is called a condensate.

124 Actually, this numerical strategy consists in condense neighboring mixed
 125 cells in one direction of the Cartesian mesh, in which interfaces are con-
 126 sidered as mono dimensional objects, namely they are considered vertical

127 during x direction step and horizontal during y direction step. A condensate
 128 then contains layers of successive different materials that are separated by
 129 straight interfaces. The thickness of these layers is calculated through vol-
 130 ume conservation. The ordering of layers is given by the $2D$ description from
 131 the previous time step. It is determined thanks to the volume fractions of
 132 neighboring cells. The layer evolution is calculated in a Lagrangian fashion
 133 which implies that layers can be as thin as partial volumes are small. Once
 134 quantities and interface positions inside the condensate are known at time
 135 t^{n+1} , they are remapped back onto the original Eulerian mesh. Finally a 2D
 136 normal in each mixed cell is computed as described in [10]: the method is
 137 based on an approximation of the gradient of the volume fraction function
 138 in mixed cells. It provides the normal to materials interface in each cell
 139 that is further used to locate materials within mixed cells. The numerical
 140 scheme used in a condensate is presented in great details in [5] and we omit
 141 this description in this work and rather focus on the interface reconstruction
 142 method.

143 As shown in [5] this numerical method has several attractive properties
 144 as conservation and perfect sliding of materials as instance. Moreover Δt is
 145 not restricted by small partial volume thanks to a tight control of density
 146 and pressure [3]. The numerical experiments carried out in [7, 2, 5, 4] have
 147 confirmed the efficiency of such a method for compressible multi-material
 148 computation. Although very promising, the method suffers from the way
 149 interfaces are dealt with.

150 In order to illustrate the interface reconstruction method NIP let us con-
 151 sider a square like interface cutting the Eulerian cells, as in Figure 2-(A).

152 These interfaces are indeed defined by their normals within each cell. NIP
 153 method consists of the following steps assuming the condensate is in the x
 154 direction:

- 155 • *Representation* Figure 2-(B). The representation step can be seen as
 156 the way of determining on which side (left or right) of the mixed cell
 157 the material is to be put. This is done by comparing the direction of
 158 the interface normal at time t^n with the vertical direction.
- 159 • *Condensate construction* Figure 2-(C). The construction of the conden-
 160 sate consists in discarding any cell edges in the mixed cells considered.
 161 Then the partial volumes of the same contiguous materials are glued
 162 together into so called condensate layers. As instance cell 2 and 3 dark
 163 materials are merged into one stand-alone layer with associated volume
 164 averaged values.
- 165 • *Condensate evolution* Figure 2-(D). The condensate layers evolution
 166 is computed from t^n to t^{n+1} thanks to the numerical scheme devel-
 167 oped in [5]. In short, each vertical interface is assigned a velocity and,
 168 consequently, a new position of each layer within the condensate is de-
 169 termined in a Lagrangian way. Any conserved variable is computed
 170 accordingly.
- 171 • *Reconstruction* Figure 2-(E). This phase consists in “guessing” the
 172 shape of each layer in the condensate before remapping. The recon-
 173 struction phase was not originally considered as a true phase of the
 174 algorithm as the author used the same shapes as the ones produced in
 175 phase *Condensate construction*, i.e. only vertical interfaces.

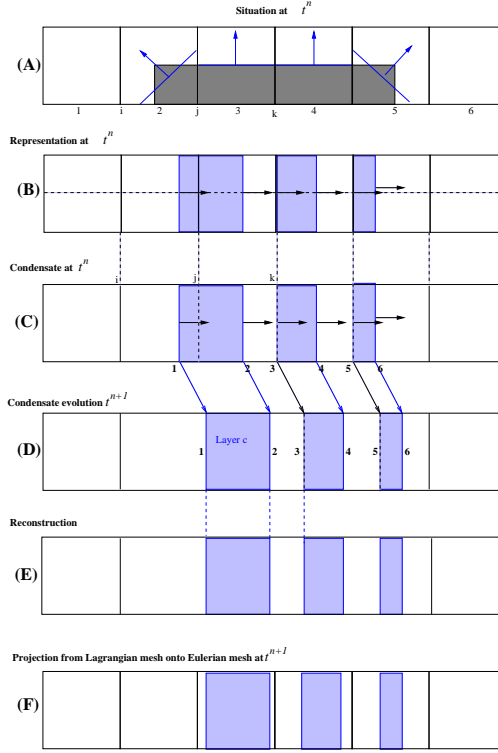


Figure 2: NIP method — (A) Situation at t^n with real materials geometry, interfaces and normals to them. (B) Representation of partial volumes at t^n . (C) Construction of a condensate at t^n by merging layers of contiguous partial volumes of the same material. (D) Evolution of condensate in a Lagrangian fashion during Δt . (E) Condensate reconstruction at t^{n+1} . (F) Condensate projection/remapping from Lagrangian mesh onto original mesh.

176 • *Projection* Figure 2-(F). The projection step consists in remapping the
 177 shapes obtained from the reconstruction phase onto the Eulerian grid.
 178 This step produces updated partial volumes in mixed cells. Volume
 179 fractions are deduced.

180 When all mixed cells in the domain are treated for direction x , the interface
 181 normals are computed using the updated volume fractions. This concludes
 182 the system evolution in direction x , as we are back to a similar situation as
 183 the one described in Figure 2-(A).

184 In the case where the normal is almost vertical, positioning the material on
 185 either side of the cell might be, at least inaccurate, or, worse, incorrect. Fur-
 186 thermore the reconstruction phase is here clearly inconsistent: the interfaces
 187 are initially horizontal in cell 3 and 4 (Figure 2-(A)), while in the Recon-
 188 struction Figure 2-(E) and in the Projection Figure 2-(F) phase interfaces
 189 are set vertical for any initial geometry. This situation of a horizontal inter-
 190 face is the worst case, but it illustrates the lack of geometrical consistency of
 191 NIP. This inaccurate reconstruction step leads to a lack of accuracy of the
 192 volume fractions obtained after the remapping step. Ultimately, it impacts
 193 the whole numerical method in any advection process.

194 As an illustration let us consider the diagonal advection of a square back and
 195 forth as shown in Figure 3. We omit the exhaustive description of this test
 196 as it will be done in the numerical Section of this paper. On the right panel
 197 it is obvious that the shape of the square is not well approximated. More
 198 important the horizontal and vertical edges of the square do not remain so.
 199 This behaviour is less pronounced if one refines the mesh but still remains.
 200 Our goal is to improve the reconstruction step so that the new method, de-

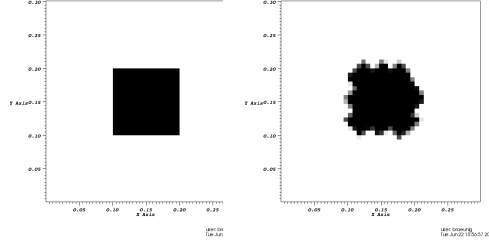


Figure 3: Diagonal advection of a square by NIP method — Left: Exact configuration — Right: Numerical configuration obtained by NIP.

201 noted ENIP standing for Enhanced Natural Interface Positioning, cure this
 202 geometrical inconsistency during the advection phase of the algorithm.

203 2.4. *FVCF-ENIP*

204 The main idea of the new interface reconstruction method ENIP emanates
 205 from the following remarks:

- 206 1. At time t^n any interface normal in mixed cell i denoted \vec{n}_i is known. It
 207 is used to locate the partial volumes within cell i when the condensate
 208 is constructed (phase (B) and (C) of Figure 2). However \vec{n}_i is never
 209 taken into account in the reconstruction and projection phases (E) and
 210 (F) from the same figure.
- 211 2. Any layer of the condensate evolves as a Lagrangian object in the orig-
 212 inal method. Consequently the cell faces could evolve in an almost
 213 Lagrangian manner within this condensate. This makes possible to
 214 conserve the initial geometry of partial volumes during this Lagrangian
 215 motion.

216 Therefore ENIP modifies several steps of NIP as depicted in Figure 4. Once
 217 a patch of neighbor mixed cells in x direction² are agglomerated, The same
 218 five steps as for NIP method are performed. The first two steps are kept
 219 unmodified. The last three are modified as described in the following.

220 2.4.1. Lagrangian Condensate evolution step

Cell interface Lagrangian velocity. After the condensate at t^n is constructed, each layer labeled c is located thanks to the left and right interface position respectively called x_c^-, x_c^+ . The numerical scheme provides the layer evolution, and as a by-product, the velocity of these interface positions, u_c^-, u_c^+ are given by

$$x_c^{-,n+1} = x_c^- + \Delta t u_c^-, \quad x_c^{+,n+1} = x_c^+ + \Delta t u_c^+. \quad (15)$$

We make the following fundamental linear displacement assumption: *The velocity linearly varies within any layer*, see Figure 5 for a sketch. This assumption implies that any point $x_i \in [x_c^-, x_c^+]$ characterized by its 1D barycentric coordinates

$$\lambda_i^- = \frac{x_c^+ - x_i}{x_c^+ - x_c^-}, \quad \lambda_i^+ = \frac{x_i - x_c^-}{x_c^+ - x_c^-}, \quad (16)$$

moves to location

$$x_i^{n+1} = \lambda_i^- x_c^{-,n+1} + \lambda_i^+ x_c^{+,n+1} = x_i + \Delta t (\lambda_i^- u_c^- + \lambda_i^+ u_c^+). \quad (17)$$

221 Then the point velocity is naturally set to $u_i = \lambda_i^- u_c^- + \lambda_i^+ u_c^+$. Using this
 222 previous formula one can associate a “Lagrangian” velocity to any cell inter-
 223 face. As instance in Figure 4-(C) cell interface located at x_i^n moves to the

²The y direction is treated likewise.

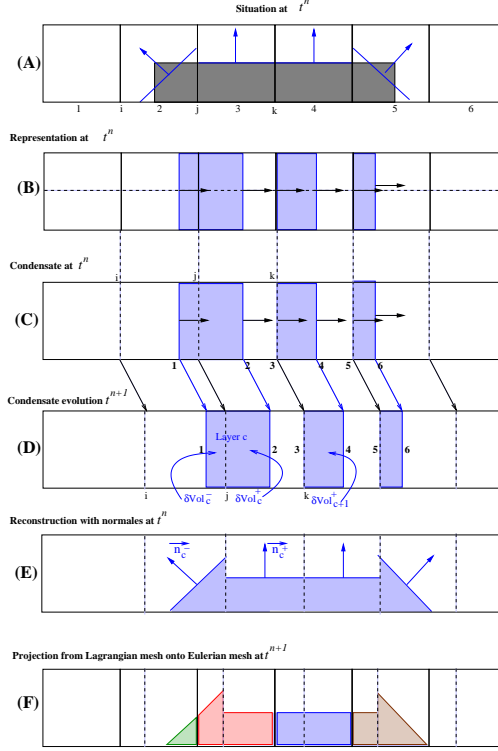


Figure 4: ENIP method — **(A)** Situation at t^n with real material, interfaces and normals to them. **(B)** Representation of material at t^n . **(C)** Construction of a condensate at t^n by merging of mixed cells leading to layers of contiguous pieces of the same material. **(D)** Evolution of condensate in a Lagrangian fashion during Δt . Determine layer compression rates δVol_c^\pm through the evolution of Lagrangian cells during Δt . **(E)** Condensate reconstruction at t^{n+1} using interface normals defined at t^n . **(F)** Condensate projection/remapping from Lagrangian mesh onto Eulerian mesh.

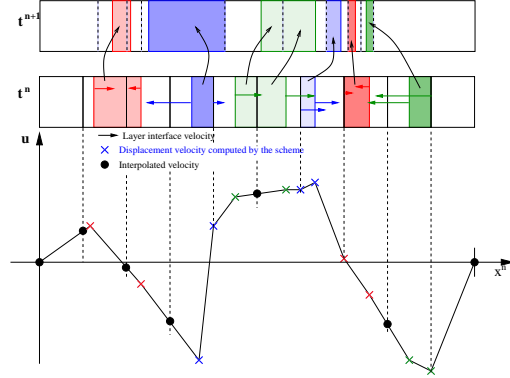


Figure 5: Sketch of linear displacement assumption — Displacement velocity varies linearly between the layer interface velocities (\times in color) computed by the numerical scheme. The cell interface velocity (\bullet) is interpolated. The two top rows represent the evolution of a condensate in the x direction from t^n to t^{n+1} .

224 position $x_i^{n+1} = x_i^n + \Delta t u_i$ with u_i being the linear combination between
 225 u_c^- and u_c^+ via the barycentric coordinates of point x_i in $[x_c^-; x_c^+]$. With the
 226 same formula one gets $x_{i+1}^{n+1} = x_{i+1}^n + \Delta u_{c+1}^-$ in the next layer as $x_{i+1}^n \equiv x_{c+1}^-$.

Compression/expansion rates. The global rate of compression/expansion in layer c during Δt is given by

$$\delta \text{Vol}_c = \frac{x_c^{+,n+1} - x_c^{-,n+1}}{x_c^+ - x_c^-} = 1 + \Delta t \frac{u_c^+ - u_c^-}{x_c^+ - x_c^-}. \quad (18)$$

The linearity assumption provides a simple way to determine the rates of compression/expansion at left/right of a point $x_i \in [x_c^-; x_c^+]$

$$\delta \text{Vol}_c^- = \frac{x_i^{n+1} - x_c^{-,n+1}}{x_c^+ - x_c^-}, \quad \delta \text{Vol}_c^+ = \frac{x_c^{+,n+1} - x_i^{n+1}}{x_c^+ - x_c^-}, \quad (19)$$

that fulfil $\delta \text{Vol}_c^- + \delta \text{Vol}_c^+ = \delta \text{Vol}_c$. Moreover the substitution of x_i^{n+1} in the previous equations yields

$$\delta \text{Vol}_c^- = \frac{x_i - x_c^-}{x_c^+ - x_c^-} + \Delta t \frac{u_i - u_c^-}{x_c^+ - x_c^-} = \lambda_i^+ + \Delta t \frac{u_i - u_c^-}{x_c^+ - x_c^-}, \quad (20)$$

where $u_i - u_c^- = (\lambda_i^- u_c^- + \lambda_i^+ u_c^+) - u_c^- = \lambda_i^+ (u_c^+ - u_c^-)$, therefore the compression rates simply writes

$$\delta \text{Vol}_c^- = \lambda_i^+ \left(1 + \Delta t \frac{u_c^+ - u_c^-}{x_c^+ - x_c^-} \right) = \lambda_i^+ \delta \text{Vol}_c, \quad (21)$$

$$\delta \text{Vol}_c^+ = \lambda_i^- \left(1 + \Delta t \frac{u_c^+ - u_c^-}{x_c^+ - x_c^-} \right) = \lambda_i^- \delta \text{Vol}_c. \quad (22)$$

Each δVol_c^+ or δVol_c^- is associated to a unique Eulerian cell; as instance in Figure 4, δVol_c^- is associated to cell 2, δVol_c^+ to cell 3, $\delta \text{Vol}_{c+1}^+$ to cell 4 and so on. Therefore δVol_c^\pm provides *de facto* the compression/expansion of the partial volume originating from its associated Eulerian cell motion. Furthermore, as any Eulerian mixed cell i possesses a unique normal denoted \vec{n}_i , this last is associated to the corresponding partial volume δVol_c^\pm ; this normal is consequently labeled \vec{n}_c^\pm . These rates are then used to reconstruct the material topology into the Lagrangian cell.

2.4.2. Reconstruction step

The Lagrangian cell $i + 1/2$ at t^{n+1} the interfaces of which moved as

$$x_i^{n+1} = x_i + \Delta t u_i, \quad x_{i+1}^{n+1} = x_{i+1} + \Delta t u_{i+1}, \quad (23)$$

changed its volume as

$$\delta \text{Vol}_{i+1/2} = \frac{V_{i+1/2}^{n+1}}{V_{i+1/2}} = \frac{x_{i+1}^{n+1} - x_i^{n+1}}{x_{i+1} - x_i} = 1 + \Delta t \frac{u_{i+1} - u_i}{x_{i+1} - x_i}. \quad (24)$$

The velocity u_i depends on u_{c-1}^-, u_{c-1}^+ and u_{i+1} depends on u_c^-, u_c^+ . Moreover $u_{c-1}^+ \equiv u_c^-$ by definition.

The second fundamental assumption states that the interface normals \vec{n}_c^\pm do not change their direction during their Lagrangian evolution. The goal is to

locate the partial volume into the Lagrangian cell at t^{n+1} and construct the linear interface, knowing its normal \vec{n}_c^\pm . Necessarily this partial volume is either in contact with cell interface x_i (superscript $+$) or x_{i+1} (superscript $-$). Its volume at t^{n+1} is given by

$$V_c^{\pm,n+1} = V_c^\pm \delta \text{Vol}_c^\mp = V_c^\pm + \Delta t \lambda_i^\mp (u_c^+ - u_c^-). \quad (25)$$

236 If $V_c^{\pm,n+1} \leq V_{i+1/2}^{n+1}$ then there exists a unique line oriented by the normal \vec{n}_c^\pm
 237 and separating the cell volume into two sub-volumes $V_c^{\pm,n+1}$ and $(V_{i+1/2}^{n+1} -$
 238 $V_c^{\pm,n+1})$ respectively by the PLIC (*“Piecewise Linear Interface Construction”*
 239 [10]) method. As the displacement velocity $u(x)$ is supposed to be piecewise
 240 linear (by the first assumption see Figure 5), then, if $x_i < x_c^- < x_{i+1}$ one
 241 deduces $x_i^{n+1} < x_c^{-,n+1} < x_{i+1}^{n+1}$. Therefore the sub-volume at t^{n+1} is strictly
 242 included into the Lagrangian cell volume $V_{i+1/2}^{n+1}$. This phase is depicted in
 243 Figure 4-(E)

244 2.4.3. Projection step

245 The projection step performs the exact intersection between the La-
 246 grangian condensate obtained after the reconstruction step in Figure 4-(E)
 247 and the Eulerian mesh (bold line squares in Figure 4-(A)). This step is de-
 248 picted in Figure 4-(F). The exact intersection consists in projecting each
 249 partial volume that is accurately located into the condensate, onto some Eu-
 250 lerian fixed cell(s). As instance in Figure 4-(F) the first partial volume is
 251 projected onto Eulerian cells 2 (green cell) and 3 (red cell). Contrarily the
 252 last partial volume is totally projected into Eulerian cell 5 (brown cell). This
 253 projection provides the quantity of material per Eulerian cell, or, equivalently
 254 its volume fraction.

255 Once volume fractions in the mixed cells are updated through the evolution
 256 of condensates, 2D normals are computed using the same technique as in
 257 original NIP method.

258 3. Numerical results

259 In this Section we present a set of test cases to assess the efficiency of
 260 the approach described in the previous Sections. First, one validates the
 261 technique on pure advection test cases that often present excessive smearing
 262 of interfaces due to the numerical inaccuracy embedded into the scheme.
 263 A square shaped object is advected with constant velocity in a diagonal
 264 direction in a first test, then into a rotating flow. Finally an hydrodynamics
 265 test case is presented.

266 3.1. Advection context

An initial square $[0.1; 0.1] \times [0.2; 0.2]$ is located into the domain $\Omega = [0 : 0.4] \times [0; 0.6]$. The density into the square is set to $\rho_0(x) = 1$ whereas it is set to $\rho_0(x) = 0$ outside. In the pure advection context this square shape should be perfectly conserved through the equation

$$\frac{\partial}{\partial t}\rho + u\frac{\partial}{\partial x}\rho + v\frac{\partial}{\partial y}\rho = 0, \quad (26)$$

where (u, v) is a constant velocity field. The exact solution at any point x and any time t is $\rho^{ex}(x, y, t) = \rho_0(x - u t, y - v t)$. If the numerical method provides an approximated solution called ρ_i^n in cell i at time t^n then the error in L_α norm is evaluated by ($\alpha = 1, 2$)

$$\varepsilon_\alpha = \frac{\sum_i |\rho_i^n - \rho^{ex}(x_i, t^n)|^\alpha}{\sum_i |\rho^{ex}(x_i, t^n)|^\alpha}. \quad (27)$$

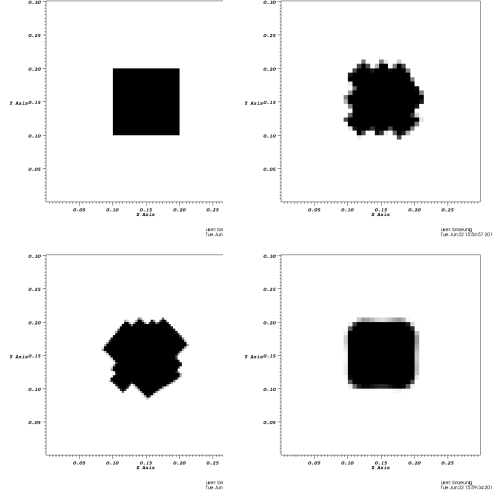


Figure 6: Advection of a square (zoom around the exact position of the initial and final square) — Top-left: exact solution — Top-right: classical NIP with a 60×60 mesh — Bottom-left: classical NIP with a 120×120 mesh — Bottom-right: ENIP with a 60×60 mesh.

267 The first test consists in advecting the square with the constant velocity
 268 field $u = 1$, $v = 3$ up to the time $t = 0.1$ then reversing the advection
 269 field by setting $u = -1$, $v = -3$ up to final time $t = 0.2$ so that the final
 270 configuration perfectly fits the initial one. Any method (NIP and ENIP
 271 included) introduces some error that we intend to measure with this test.
 272 In Figure 6 are shown the exact solution (top-left) and the results obtained
 273 with a 60×60 mesh for NIP (top-right) and ENIP (bottom-right). ENIP is
 274 visibly able to preserve the shape of the square whereas NIP is not. A mesh
 275 refinement of NIP computation (120×120 mesh for the bottom-left panel)
 276 does not improve the situation. In table 1 we gather the errors for the L_1 ,
 277 L_2 norms for successively refined meshes for the NIP and the proposed ENIP

$\Delta x = \Delta y$	L_1 NIP	L_1 ENIP	L_2 NIP	L_2 ENIP
0.02	3.652	0.196	2.575	0.079
0.0133	0.389	0.165	0.318	0.081
0.01	0.339	0.111	0.284	0.053
0.005	0.221	0.042	0.195	0.017
0.0033	0.155	0.025	0.138	0.010

Table 1: Error in L_1, L_2 norms for the advection problem — NIP versus ENIP methods.

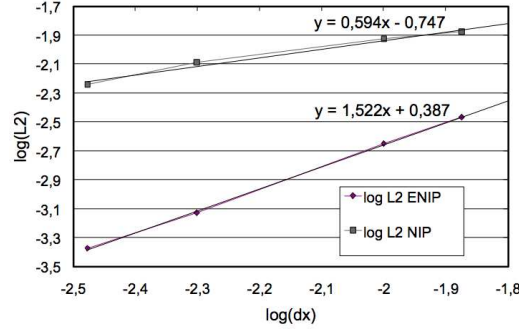


Figure 7: Convergence of ENIP vs NIP for a pure advection problem. The log of the L_2 error is displayed as a function of the log of Δx .

278 method on this advection problem. Systematically ENIP over-tops NIP. In
279 Figure 7 we display the log-log scale results for the error in L_2 norm for both
280 methods showing the improvement gained by ENIP; indeed the slope which
281 represents a measure of the numerical order of convergence is improved by a
282 factor 2.5 (0.6 for NIP and 1.5 for ENIP).

The next test consists in the rigid rotation of a square $[0.06; 0.46] \times [0.3; 0.7]$ (density 1) into the unit square domain, see Figure 8 top-left panel. A 100×100 uniform mesh is considered and the rotation is given by the

Figure 8: Top-left: Sketch of the rigid rotation of a square — Top-right: After 5/8 of the full rotation — Bottom-left: After one full rotation — Bottom-right: After three full rotations.

velocity field

$$u = -100(y - 0.5), \quad v = 100(x - 0.5).$$

283 In Figure 8 we display the density after 5/8 of the full rotation, after one
 284 and three rotations. The square shape is almost preserved. Contrarily the
 285 classical NIP method would totally lose the shape after one rotation.

286 3.2. *Hydrodynamics context*

287 We run an idealized 2D test case that corresponds to the free drop of
 288 a liquid rectangle within a 2D rectangular tank filled with gas [4]. This

289 context is inspired by the problem of sloshing that may appear in the tanks
 290 of Liquid-Natural-Gas (LNG) carriers. The study focuses on the ability for
 291 the numerical simulations to take properly into account the physics that is
 292 of major importance during the liquid impact such as the escape of the gas
 293 underneath and its compression. As a strong sliding process occurs between
 294 the compressed gas and the falling liquid. The ability of the method to
 295 properly deal with sliding conditions at the interface has a major effect on
 296 the final numerical compression and shape of the trapped air. This has
 297 ultimately a strong influence on the impact pressure.

298 The test case consists in a domain $\Omega = [0.0; 0.0] \times [10m; 15m]$ filled with air.
 299 The liquid is initially at rest in the rectangle $[0; 2] \times [5; 10]$ and is falling under
 300 the gravity that is pointing downward with magnitude $g = 9.81m.s^{-2}$. A
 301 free fall of the liquid into vacuum would impact at $t_{\text{impact}} = 0.64s$ however
 302 due to the presence of the gas this theoretical value is not correct for our
 303 simulation however some critical phenomena still occur in the vicinity of
 304 this time. As instance around t_{impact} a pocket of gas is trapped under
 305 the falling liquid and this strongly impacts the numerical impact pressure
 306 by decelerating and damping the free fall of the liquid. Therefore a good
 307 interface reconstruction method should qualitatively improve the numerical
 308 results. One considers a mesh made of 100×150 uniform cells on the domain.

309 One shows the results for NIP and ENIP at time $t = 0.6s$ Figure 9-(a)-(b)
 310 and $t = 0.64s$ in Figure 9-(c)-(d). The classical NIP method was already able
 311 to deal with such sliding effects. However the interface reconstruction method
 312 employed is not accurate and stable enough to be free of oscillation that
 313 one suspects to be only a numerical artifacts (see panels (a-c)). Contrarily

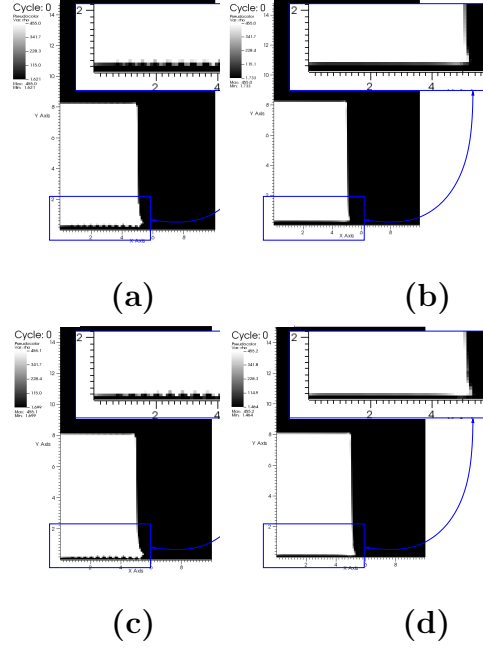


Figure 9: — (a) NIP results at $t = 0.6$ (full view and zoom on the impact zone) — (b) ENIP results at $t = 0.6$ — (c) NIP results at $t = 0.64$ — (d) ENIP results at $t = 0.64$.

the new reconstruction method ENIP on this very same test case is able to produce a smooth interface that permits to obtain a more realistic simulation. Indeed this simulation prominently displays the fact that the “bubbling” effects of NIP is of pure numerical origin and that ENIP cures this drawback.

4. Conclusion and perspectives

This paper deals with the improvement of the so-called NIP (Natural Interface Positioning) method. The NIP method described in [5] is an add-on to the FVCF method in order to treat multi-material fluid flows uses the

323 concept of condensate. A condensate is the association of contiguous mixed
 324 cells in either x or y direction. They are further treated as an entity to
 325 make possible the treatment of each mixed cell taken individually. NIP is
 326 the method based on the following steps: *Representation*, *Condensate con-*
 327 *struction*, *Condensate evolution*, *Reconstruction*, and *Projection*. The present
 328 paper points the weakness of the NIP method in pure advection context and,
 329 consequently, in a full multi-material hydrodynamics one. An enhanced NIP
 330 method is proposed (ENIP). It modifies several of the previous listed steps.
 331 More precisely the condensate is assumed to evolved in an almost-Lagrangian
 332 fashion. The reconstruction step assumes that the condensate keeps the same
 333 form modulo some expansion/compression that the numerical scheme already
 334 provides. So the displacement of the condensate is performed either with the
 335 true computed velocity or with an interpolation of it. *In fine* the condensate
 336 preserves its topology contrarily to the original NIP method for which the
 337 condensate has no recollection of its shape from the beginning of the time
 338 step.

339 The capability of the full numerical method is now dramatically improved
 340 as seen on advection test cases (advection and rigid rotation of a square).
 341 Moreover we ran ENIP on a difficult mutli-material hydrodynamics tests
 342 simulating the free drop of a liquid rectangle within a 2D rectangular tank
 343 filled with gas in the context of sloshing that may appear in the tanks of
 344 Liquid-Natural-Gas carrier (see [4]). The accuracy, stability and robustness
 345 of the ENIP method is clearly seen especially at the time some air is trapped
 346 under the water. In the near future we plan to investigate the evolution of
 347 this method to the case of mixed cells with more than two materials. In this

348 case the only difficulty lays in the positioning of the different materials in
 349 the cell, but their evolution within the condensate follows exactly the same
 350 algorithm ENIP with no modification of the numerical scheme. We also plan
 351 to investigate the evolution of the method in 3D.

352 References

- 353 [1] D.M. Anderson, G.B. McFadden, A.A. Wheeler, A Diffuse-Interface
 354 Methods in Fluid Mechanics, *Ann. Rev. Fluid Mech.*, Vol. 30, p. 139–
 355 165, (1998).
- 356 [2] J.-P. Braeunig, Sur la simulation d'écoulements multi-matériaux par une
 357 méthode eulérienne directe avec capture d'interfaces en dimensions 1, 2
 358 et 3 (technical part in english), Thèse de Doctorat N° 2007/85, Ecole
 359 Normale Supérieure de Cachan, (2007).
- 360 [3] J.P. Braeunig, An algorithm to control the pressure evolution for the
 361 FVCF-NIP method for compressible multi-material fluid flows *Int. J.*
 362 *Finite Vol.* in press (2010).
- 363 [4] J.P. Braeunig, L. Brosset, F. Dias, J.-M Ghidaglia, Phenomenological
 364 study of liquid impacts through 2D compressible two-fluid numerical
 365 simulations, *19th Int. Offshore and Polar Eng. Conf. Osaka, Japan,*
 366 *ISOPE* (2009)
- 367 [5] J.P. Braeunig, B. Desjardins, J.-M Ghidaglia, A totally Eulerian fi-
 368 nite volume solver for multi-material fluid flows, *European Journal of*
 369 *Mechanics B/Fluids*, vol. 28, No4, pp. 475-485 (2009).

- 370 [6] V. Dyadechko, M. Shashkov, Reconstruction of multi-material interfaces
371 from moment data. *J. Comput. Phys.*, 227, pp. 5361-5384, (2008).
- 372 [7] J.-M. Ghidaglia, A. Kumbaro, G. Le Coq, On the numerical solution to
373 two fluid models via a cell centered finite volume method, *Eur. J. Mech.*
374 *B-Fluids*, Vol. 20, No 6, p.841–867, (2001).
- 375 [8] B. Lefaurie C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Mod-
376 elling merging and fragmentation in multifluid flows with SURFER. *J.*
377 *Comput. Phys.*, Vol. 113, no. 1, p. 134–147, (1994).
- 378 [9] W.F. Noh, P. Woodward, SLIC (Simple Line Interface Calculation), Lec-
379 tures notes in Physics 59. *Editions Springer*, Berlin,(1976).
- 380 [10] D.L. Youngs, Time-Dependent multi-material flow with large fluid dis-
381 tortion, *Numerical Methods for Fluid Dynamics*, edited by K. W. Morton
382 and M. J. Baines, p. 273–285, (1982).