



**HAL**  
open science

## Fictitious domain methods for two-phase flow energy balance computations in nuclear components

Michel Belliard, Isabelle Ramière

► **To cite this version:**

Michel Belliard, Isabelle Ramière. Fictitious domain methods for two-phase flow energy balance computations in nuclear components. *International Journal for Numerical Methods in Fluids*, 2012, 68 (8), pp.939-957. hal-00604069

**HAL Id: hal-00604069**

**<https://hal.science/hal-00604069>**

Submitted on 4 Jul 2011

**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.

# Fictitious domain methods for two-phase flow energy balance computations in nuclear components

M. Belliard, I. Ramière

CEA, DÉPARTEMENT D'ÉTUDES DES RÉACTEURS, DEN, DER/SSTH/LDAL  
CEA, DÉPARTEMENT D'ÉTUDES DES COMBUSTIBLES, DEN, DER/SESC/LSC  
Cadarache, F-13108 Saint Paul-lez-Durance Cedex, France

July 4, 2011

## Abstract

This paper is dedicated to the numerical simulation of nuclear components (Cores and Steam Generators-SG) by fictitious domain methods. The fictitious domain approach consists in immersing the physical domain under study in a Cartesian domain, called the fictitious domain, and in performing the numerical resolution on this fictitious domain. The calculation times are then efficiently reduced by the use of fast solvers. In counterpart, one has to handle with an immersed boundary, generally non-aligned with the Cartesian mesh, which can be non-trivial.

The two fictitious domain methods compared here on industrial simulations and developed by Ramière and coworkers, deal with an approximate immersed interface directly derived from the uniform Cartesian mesh. All the usual immersed boundary conditions (Dirichlet, Robin, Neumann), possibly mixed, are handled through a unique formulation of the fictitious problem. This kind of approximation leads to first-order methods in space that exhibit a good ratio of the precision of the approximate solution over the CPU time, which is very important for industrial simulations.

After a brief recall of the fictitious domain method with spread interface (Ramière *et al.*, CMAME 2007) and the fictitious domain method with immersed jumps (Ramière *et al.*, JCP 2008), we will focus on the numerical results provided by these methods applied to the energy balance equation in a SG. The advantages and drawbacks of each

method will be pointed out. Generally speaking, the two methods confirm their very good efficiency in term of precision, convergence and calculation time in an industrial context.

Keywords : Fictitious Domain; Immersed Boundary; Immersed Interface; Nuclear component; Steam Generator; Finite Elements; Finite Volumes; Computational Fluid Dynamics.

## 1 Introduction

This paper concerns the industrial numerical simulations of nuclear components (cores and Steam Generators, SGs) by a Fictitious Domain Method (FDM) following the works of Ramière and coworkers [1, 2, 3, 4]. These works were indeed in line with the numerical R&D work package of the Neptune platform [5], which is a software platform dedicated to the thermal-hydraulic numerical simulation of nuclear power plants from the local scale to the system scale through the component scale. The thermal-hydraulic simulation of nuclear power plants consists in modeling two-phase flow (water/steam) passing into obstacles: we are hence in presence of free and blind regions with eventually free boundaries. The vibrations and displacements of the obstacles under the flow action (fluid-structure interaction) are of great importance for the safety analysis. Due to computer limited resources for industrial component simulations, the two-phase fluid is usually modeled by an equivalent mixture fluid in a porous region thanks to a homogenization process [6]. This mixture fluid verifies Navier-Stokes-like balance equations with specific non-linear coefficients and source terms. Based on this formalism, the CEA (which is the French Atomic Commission) has developed two softwares, Genepi [7] and Flica-IV [8], devoted to the engineering SG and core simulations respectively. Numerically speaking, the Genepi software uses a Finite Element (FE) resolution while the Flica-IV software is based on a Finite Volume (FV) discretization.

Steam generators of nuclear power plants produce the steam feeding the turbines. Those material can be roughly described as a 10 m high and 3 m diameter cylinder. The riser is filled with about 5,000 upside down U-shaped tubes of 1 cm diameter in which flows hot water (primary fluid) coming from the nuclear core, see Figure 1. Some technological objects (e.g. tube-support plates) complete the riser. The liquid feed water (secondary fluid) entering in the riser is then heated by contact with the hot upside down U-tube bundle. That leads to a liquid/steam fluid. After liquid separation, the 'dry' steam flows into the electricity turbines. We distinguish the so-called 'hot leg' (on

the primary fluid inlet side) and 'cold leg' (on the primary fluid outlet side) of the SG.

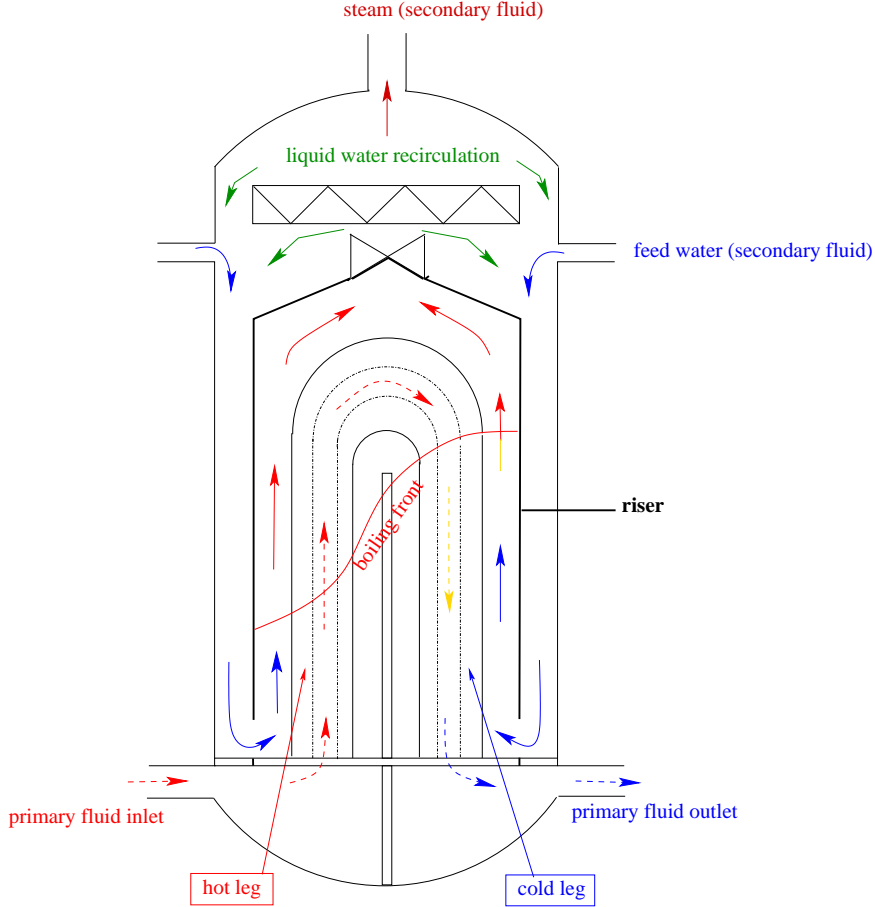


Figure 1: Simplified scheme of a steam generator.

Here, we focus on the SG's energy balance equation. As previously mentioned, the two-phase secondary fluid is viewed as an equivalent mixture fluid [7]. Hence the secondary fluid energy balance equation is a convection-diffusion-like with non-linear source terms

$$\beta \partial_t(\rho H) + \text{div}(\beta \mathbf{G} H) - \text{div}(\beta \chi_T \nabla H) = \beta Q - \text{div}(\beta x(1-x)\rho \mathcal{L} \mathbf{V}_R). \quad (1)$$

where the unknown is the mixture specific enthalpy  $H$  ( $KJ.kg^{-1}$ ) and

- $\beta$  is the porosity ( $\equiv \omega_m/\omega$ ) with  $\omega$  the homogenization cell volume ( $m^3$ ) and  $\omega_m$  the mixture volume (liquid + steam) in the homogenization cell ( $m^3$ ),

- $\rho(H, P)$  is the density ( $kg.m^{-3}$ ),
- $P$  is the mixture pressure ( $Pa$ ),
- $G$  is the mixture mass flux ( $\equiv \rho \mathbf{V}$ , with  $\mathbf{V}$  the mixture velocity ( $m.s^{-1}$ )),
- $\chi_T = a|\mathbf{G}|L$  is the turbulent diffusion coefficient ( $kg.m^{-1}.s^{-1}$ ) [9] with  $L$  a typical eddy length and  $|\cdot|$  the Euclidean norm,
- $Q$  is the heat source term ( $W.m^{-3}$ ),
- $\mathcal{L}(P)$  is the latent heat of vaporization ( $J.kg^{-1}$ ),
- $x(H, P)$  is the static quality ( $\equiv \frac{H - H_{ls}}{\mathcal{L}}$ ) with  $H_{ls}(P)$  the saturation liquid enthalpy ( $KJ.kg^{-1}$ ),
- $\mathbf{V}_R$  is the relative velocity (steam velocity minus liquid velocity,  $m.s^{-1}$ ) [10, 11].

The  $\beta x(1-x)\rho\mathcal{L}\mathbf{V}_R$  term is called the drift term. The mixture pressure  $P$  and mass flux  $\mathbf{G}$  are data of the problem (generally obtained by the resolution of the Navier-Stokes balance equations). The heat source term  $Q$  is given by the resolution of the primary fluid energy balance equation in which external correlations involving  $H$  and  $\mathbf{G}$  are used.

In most of industrial simulations, the shape of the physical domain involves numerical resolution methods based on unstructured body-fitted meshes. These classical methods induce a loss of efficiency and rapidity of numerical solvers in comparison with methods based on Cartesian meshes. Indeed, with Cartesian-based methods, the natural tensor formulation can be used, fast solvers (based for instance on finite volume methods) and multi-level or multi-resolution techniques (see [12, 13] for example) can be easily implemented, ... Moreover, in the particular case of moving boundaries, the cost of mesh generation and re-meshing for boundary-fitted meshes can be significant. In this context, FDMs are full of promising as they consist in uncoupling the physical domain of interest and the computational domain. This computational domain includes the physical domain and its moving and is generally chosen to be easily meshed with a Cartesian grid. We then enjoy all the advantages of Cartesian-based methods listed below without managing the re-meshing.

Concerning nuclear component simulations, the fictitious domain concept allows furthermore the use of a unique formalism for free, porous or blind

regions inside the component [14], with fixed or moving boundary (fluid-structure interactions, free-surface flows in case of liquid leak, ...).

In this paper, our goal is to show the capability of an industrial software, initially based on body-fitted unstructured meshes, to take advantage of the fictitious domain approach without large modifications of this software.

In Sections 2 and 3, we detail the fictitious domain philosophy and more particularly the two fictitious domain methods introduced in [2] and [3] respectively, which seem promising in an industrial context such the one described here. In Section 4, we are interested in solving the SG's energy balance equation by these two fictitious domain methods. More precisely, we investigate the comparison in terms of accuracy and computer efficiency of the fictitious domain methods and the classical FE body-fitted mesh solver usually employed in engineering simulations. Finally some words of conclusion are given.

## 2 The fictitious domain concept

In the fictitious domain approach, the original domain  $\tilde{\Omega}$  is embedded in a fictitious domain  $\Omega$  which is geometrically bigger and generally simpler-shaped (cf. Figure 2). Doing this, some immersed boundary  $\Sigma$  appears such that  $\Omega = \tilde{\Omega} \cup \Sigma \cup \Omega_e$ , where  $\Omega_e$  is the complementary domain (often called 'exterior domain'). The spatial discretization is now performed in  $\Omega$ , independently of the shape of the original domain  $\tilde{\Omega}$ . Numerical methods involving structured or Cartesian meshes can be used. Consequently, the resolution of the new problem in  $\Omega$  will be faster and simpler.

The main issue is to enforce the original boundary conditions (B.C.) on the immersed interface  $\Sigma$  which is non-aligned with the mesh. Indeed, the restriction of the fictitious solution over the physical domain is expected to be an accurate approximation of the physical solution.

The first fictitious domain methods were introduced in the fifties by Hyman [15] and the Russian's school [16, 17, 18, 19, 20] and for a few years now, fictitious domain methods have been hugely developed and have arisen in different fields: computational fluid dynamics (e.g. [14]), solid mechanics (e.g. [21]), medical simulation (e.g. [22]), ... The FDMs can be classified into two main categories depending on the way they enforce the immersed boundary conditions. In the first category, the complementary domain has only a numerical purpose and can be used to impose the immersed boundary conditions: we design this group of methods as *Immersed boundary methods*. In this category, we recover the root idea of the precursor fictitious domain

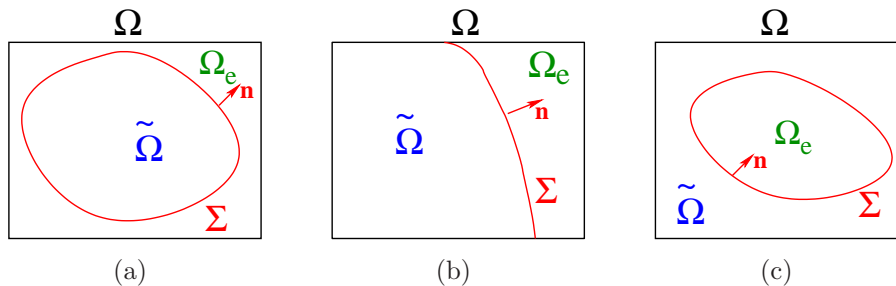


Figure 2: Embedding the original domain  $\tilde{\Omega}$  inside a fictitious rectangular domain  $\Omega$ .

methods of the Russian's school. The second category contains methods based on immersed transmission conditions (jump conditions) that link two physical domains. The complementary domain has then a real physical sense. This category is called *Immersed interface methods*. As the transmission conditions can be arranged to manage immersed boundary conditions, methods of this category can generally also be derived in immersed boundary methods ([23, 24, 25]).

In a non-exhaustive way, we can mention the following FDMs for each of the previous categories.

- *Immersed boundaries* (between one physical domain and one supplementary domain): the Immersed Boundary Method (I.B.M.) [26, 27], the truncated domain method or cut-cell method [28, 29], the Direct Forcing method [30, 31], the fictitious domain methods with surface Lagrange multipliers [32, 33] or the distributed (volume) Lagrange multipliers [34], the penalty methods [16, 35, 36], the Finite Cell Method [21], the Ghost-Cell method [37], the Fat Boundary Method [38, 39], the fictitious domain with spread interface [2], the diffuse domain approach [40], the embedded finite-difference method [41], the X-FEM-based fictitious domain methods [42, 43],...
- *Immersed interfaces* (between two physical domains): the Immersed Interface Method (I.I.M.) [44, 45], the Ghost-Fluid method [46], the Matched Interface and Boundary method (M.I.B.) [47], the Algebraic Immersed Interface and Boundary method [48], the fictitious domain method with immersed jumps [49, 3],...

Very few methods enable us to deal with all usual embedded boundary conditions (Dirichlet, Neumann or Robin), possibly mixed, within a unique formulation. The two FDMs detailed in the next section offer this possibility

which is very comfortable, especially for an industrial software that has to deal with all the possible physical situations.

### 3 Two general fictitious domain methods

In this section, we present the main characteristics of the fictitious domain method with spread interface [2], which is an immersed boundary method, and of the fictitious domain method with immersed jumps [3], which is an immersed interface method. For more technical details, the reader is invited to consult the corresponding articles.

We are interesting in solving a convection-diffusion-reaction-type equation on a physical open bounded domain  $\tilde{\Omega} \subset \mathbb{R}^d$  with a boundary  $\partial\tilde{\Omega}$  that is sufficiently regular. In its general form, the original problem  $(\tilde{\mathcal{P}})$  writes: For  $\tilde{\mathbf{a}} \in (L^\infty(\tilde{\Omega}))^{d \times d}$ ,  $\tilde{\mathbf{v}} \in L^\infty(\tilde{\Omega})^d$ ,  $\tilde{b} \in L^\infty(\tilde{\Omega})$  and  $\tilde{f} \in L^2(\tilde{\Omega})$ , find  $\tilde{u} \in H^1(\tilde{\Omega})$  such that

$$(\tilde{\mathcal{P}}) \begin{cases} \operatorname{div}(-\tilde{\mathbf{a}} \cdot \nabla \tilde{u} + \tilde{\mathbf{v}} \tilde{u}) + \tilde{b} \tilde{u} = \tilde{f} & \text{in } \tilde{\Omega} \\ \text{B.C.} & \text{on } \partial\tilde{\Omega}, \end{cases} \quad (2)$$

The symmetric tensor of diffusion  $\tilde{\mathbf{a}} \equiv (\tilde{a}_{ij})_{1 \leq i, j \leq d}$  and the reaction coefficient  $\tilde{b}$  verify the classical ellipticity assumptions. Hence the solution  $\tilde{u}$  of the problem  $(\tilde{\mathcal{P}})$  exists and is unique.

The generic aspect of the fictitious domain methods developed by Ramière and coworkers is to consider that all the general B.C., which will become immersed B.C. in a fictitious domain context, can be expressed as a Robin boundary condition:

$$-(\tilde{\mathbf{a}} \cdot \nabla \tilde{u}) \cdot \mathbf{n} = \alpha_R \tilde{u} + g_R \quad (3)$$

where  $\mathbf{n}$  is the outward unit normal vector on  $\partial\tilde{\Omega}$ ,  $0 \leq \alpha_R \in L^\infty(\partial\tilde{\Omega})$  and  $g_R \in L^2(\partial\tilde{\Omega})$ .

Indeed, a Neumann B.C. is directly obtained setting  $\alpha_R = 0$  in the previous expression while a Dirichlet B.C.  $u = u_D$  can be viewed as a penalized Robin B.C.:

$$\alpha_R = \frac{1}{\eta}, \quad g_R = -\frac{1}{\eta} u_D \quad (4)$$

where  $0 < \eta \ll 1$  is a real penalty parameter, called penalty coefficient, which is chosen to be very small (typically around  $10^{-12}$ ).

With this formalism, mixed boundary conditions (*i.e.* different B.C. supporting by different parts of the boundary) are easily taken into account by setting the suitable values of the Robin coefficients on the different parts of



the boundary.

Hence the fictitious domain methods of Ramière and coworkers focus on the modelization of immersed Robin B.C.

### 3.1 The fictitious domain method with immersed spread interface [2] (ISI)

The fictitious problem ( $\mathcal{P}^s$ ) to be solved in  $\Omega$  is build from the original problem ( $\tilde{\mathcal{P}}$ ) by extending the physical partial differential equation (PDE) to the whole fictitious domain  $\Omega$ . The coefficients of the equation in  $\Omega_e$  are *a priori* not determined. In order to handle the immersed B.C. on  $\Sigma$ , we consider the transmission problem between  $\tilde{\Omega}$  and  $\Omega_e$  preserving the continuity of the solution  $u$  on  $\Sigma$ . In the distribution sense, we then get:

$$(\mathcal{P}^s) \begin{cases} \operatorname{div}(-\mathbf{a} \cdot \nabla u + \mathbf{v}u) + bu = f - [(\mathbf{a} \cdot \nabla u) \cdot \mathbf{n}]_{\Sigma} \delta_{\Sigma} + [(\mathbf{v} \cdot \mathbf{n})]_{\Sigma} u \delta_{\Sigma} & \text{in } \Omega \\ \text{original B.C.} & \text{on } \partial\Omega \cap \partial\tilde{\Omega} \\ \text{suitable B.C.} & \text{on } \partial\Omega \setminus \partial\tilde{\Omega} \end{cases} \quad (5)$$

where the tensor of diffusion  $\mathbf{a}$  and the reaction coefficient  $b$  satisfy classical ellipticity assumptions. The symbol  $[\cdot]_{\Sigma}$  denotes the jump across  $\Sigma$  oriented by the outward unit normal  $\mathbf{n}$  (turned to the exterior of  $\tilde{\Omega}$ ), while  $\delta_{\Sigma}$  is the Dirac delta measure supported by  $\Sigma$ .

For B.C. correctly chosen on  $\partial\Omega \setminus \partial\tilde{\Omega}$ ,  $\mathbf{a} \in (L^{\infty}(\Omega))^{d \times d}$ ,  $\mathbf{v} \in L^{\infty}(\Omega)^d$ ,  $b \in L^{\infty}(\Omega)$  and  $f \in L^2(\Omega)$ , we can prove that the solution  $u \in H^1(\Omega)$  of the problem ( $\mathcal{P}^s$ ) exists and is unique.

**Remark 1** *The coefficients  $\mathbf{a}$ ,  $\mathbf{v}$ ,  $b$  and  $f$  as well as the solution  $u$  may depend on the penalty coefficient  $\eta$ . However for sake of simplicity, we omit this index in the notation.*

This method enables us to take into account immersed jumps of fluxes (diffusion and convection). However, as the continuity of the solution across the immersed interface is assumed, the treatment of a general immersed Robin B.C. requires the control of the coefficients of the fictitious problem in the complementary domain. Hence, we consider this method as an immersed boundary method.

The diffusion flux jump carried by  $\Sigma$  contains the information about the immersed Robin B.C. For sake of simplicity we impose  $-(\mathbf{a} \cdot \nabla u) \cdot \mathbf{n}|_{\Sigma}^+ \approx 0$  by setting  $\mathbf{a}_e = \eta \mathbf{Id}$  on  $\Omega_e$ , so that

$$[(\mathbf{a} \cdot \nabla u) \cdot \mathbf{n}]_{\Sigma} = \alpha_R u|_{\Sigma} + g_R \quad (6)$$

We then have to impose  $\mathbf{v}|_{\Omega_e} = \mathbf{v}_e = \mathbf{0}$  in order unconstrained the solution in the complementary domain  $\Omega_e$ .

The convection-diffusion equation to be solved in  $(\mathcal{P}^s)$  can be finally still written in the distribution sense as

$$\operatorname{div}(-\mathbf{a}\cdot\nabla u + \mathbf{v}u) + bu = f - (\alpha_R u + g_R + (\mathbf{v}\cdot\mathbf{n})^- u) \delta_\Sigma \quad (7)$$

The problem  $(\mathcal{P}^s)$  then guaranties  $u|_{\tilde{\Omega}} \rightarrow \tilde{u}$  when  $\eta \rightarrow 0$ .

Concerning the numerical aspects, if a Finite Element or a Finite Volume scheme is used, the Dirac delta measure of Eq.(7) will lead to an integral over the immersed boundary  $\Sigma$ . However, in a fictitious domain context, a Cartesian mesh is used and hence the support of  $\Sigma$  is not exactly defined. Ramière *et al.* [2] then proposed to use the characteristic function as mollifiers [50] of the Dirac measure  $\delta_\Sigma$  in order to distribute the effect of this Dirac measure to an approximate spread interface  $\omega_{h,\Sigma}$  derived from the mesh. This approximate immersed interface is simply defined as the union of the cells crossed by the immersed interface  $\Sigma$ , see Fig. 3. The jumps of fluxes introduced in Eq.(5) are then spread on  $\omega_{h,\Sigma}$  thanks to a characteristic parameter  $\epsilon_h$  that ensures the local conservativity of the fluxes. In practice, this cell parameter is the ratio of the measure of the cell by the measure of the original immersed interface intersected by this cell (see [2] for more details).

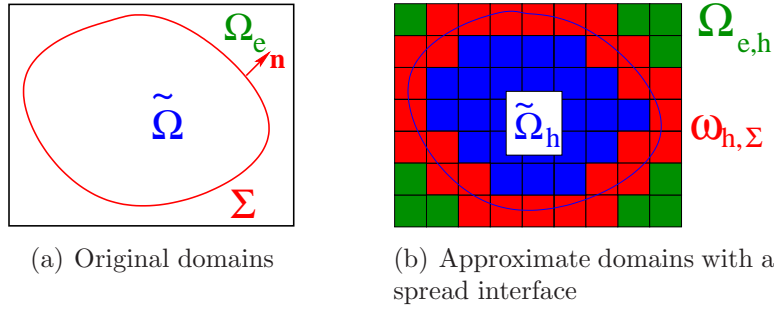


Figure 3: Spread interface  $\omega_{h,\Sigma}$  derived from the Cartesian mesh.

In the end, we solve the following generic discrete convection-diffusion fictitious problem  $(\mathcal{P}_h^s)$  over the Cartesian mesh of  $\Omega$

$$(\mathcal{P}_h^s) \begin{cases} \operatorname{div}(-\mathbf{a}_h \cdot \nabla u_h + \mathbf{v}_h u_h) + b_h u_h = f_h & \text{in } \Omega_h \\ \text{original B.C.} & \text{on } \partial\Omega_h \cap \partial\tilde{\Omega}_h \\ \text{suitable B.C.} & \text{on } \partial\Omega_h \setminus \partial\tilde{\Omega}_h \end{cases}$$

where the discrete coefficients are reported in Table 1. On  $\partial\Omega_h \cap \partial\omega_{h,\Sigma}$ , a homogeneous Neumann B.C. is imposed to be coherent with the required

external flux, see Eq. (6).

In $\tilde{\Omega}_h$	In $\omega_{h,\Sigma}$	In $\Omega_{e,h}$
$\mathbf{a}_h = \tilde{\mathbf{a}}_h, \mathbf{v}_h = \tilde{\mathbf{v}}_h,$ $b_h = \tilde{b}_h, f_h = \tilde{f}_h$	$\mathbf{a}_h = \tilde{\mathbf{a}}_h, \mathbf{v}_h = \tilde{\mathbf{v}}_h,$ $b_h = \tilde{b}_h + \frac{\alpha_{R,h}}{\epsilon_h} + \frac{\tilde{\mathbf{v}}_h \cdot \mathbf{n}}{\epsilon_h}, f_h = \tilde{f}_h - \frac{g_{R,h}}{\epsilon_h}$	$\mathbf{a}_h = \eta \mathbf{Id}, \mathbf{v}_h = \mathbf{0},$ $b_h = 0, f_h = 0$

Table 1: Coefficients of the fictitious convection-diffusion problem with spread interface

The rate of convergence of this fictitious domain method has been proved [4] to be in  $\mathcal{O}(h^{1/2})$  for the  $H^1$ -norm and in  $\mathcal{O}(h)$  for the  $L^2$ -norm, where  $h$  denotes the mesh step. In [2], various academic numerical test cases have been performed to validate this spread interface fictitious domain method. A  $Q_1$  FE scheme had been used. The results confirm the accuracy of the method for all general B.C. as well as the expected orders of convergence. The modelling error with respect to the penalty parameter  $\eta$  is also looked at in [2]: we obtain a behaviour in  $\mathcal{O}(\eta^{1/3})$  for an immersed Dirichlet problem.

**Remark 2** *When only a Dirichlet boundary condition is under consideration, this method returns to a classical volume penalization method (see e.g. [35]).*

### 3.2 The fictitious domain method with immersed jumps [49, 3] (JEBC)

The acronym JEBC used by Angot for this method refers to 'jump embedded boundary conditions' even if this method deals with more general embedded jump conditions. The first model with embedded jump transmission conditions has been proposed in [51]. Using these transmission conditions, the treatment of general embedded B.C. has been theoretical developed in [49] and numerically performed and validated in a fictitious domain context in [3]. The fictitious domain problem is an extended convection-diffusion problem with immersed jump conditions:

$$(\mathcal{P}^j) \begin{cases} \operatorname{div}(-\mathbf{a}\nabla u + \mathbf{v}u) + bu = f & \text{in } \Omega, \\ \text{original B.C.} & \text{on } \partial\Omega \cap \partial\tilde{\Omega} \\ \text{suitable B.C.} & \text{on } \partial\Omega \setminus \partial\tilde{\Omega}, \\ \llbracket (\mathbf{a}\nabla u) \cdot \mathbf{n} \rrbracket_{\Sigma} = \alpha \bar{u}|_{\Sigma} - q & \text{on } \Sigma, \\ \overline{(\mathbf{a}\nabla u) \cdot \mathbf{n}}|_{\Sigma} = \beta \llbracket u \rrbracket_{\Sigma} - g & \text{on } \Sigma \end{cases} \quad (8)$$

where the symbol  $[[\cdot]]_\Sigma$  represents the jump of traces across  $\Sigma$  oriented by the unit normal vector  $\mathbf{n}$  (turned towards  $\Omega_e$ ) while the symbol  $\overline{\cdot}$  refers to the arithmetic mean of traces on  $\Sigma$ .

The uniformly positive definite tensor of diffusion  $\mathbf{a}$ , the reaction coefficient  $b$  and the transfer coefficients  $\alpha, \beta$  (defined on  $\Sigma$ ) are supposed measurable and verifying classical ellipticity assumptions. For data  $f \in L^2(\Omega)$ ,  $g$  and  $q$  given in  $L^2(\Sigma)$  and B.C. on  $\partial\Omega \setminus \partial\tilde{\Omega}$  correctly chosen, the problem  $(\mathcal{P}^j)$  is proved to be well-posed (see [49]) and the solution  $u$  belongs to  $H^1(\tilde{\Omega} \cup \Omega_e)$ .

**Remark 3** *When  $\alpha = g = q = 0$  and  $\beta \rightarrow \infty$ , the perfect transmission problem is recovered with  $u \in H^1(\Omega)$ .*

The jump conditions on  $\Sigma$  are more general than in the I.I.M. where the jumps of solution and flux are supposed to be given. A suitable choice of the transmission conditions coefficients of the problem  $(\mathcal{P}^j)$  enable us to impose Robin immersed B.C. on  $\Sigma$  without requiring any control of the exterior problem

$$\alpha = 4\beta = 2\alpha_R, \quad g - \frac{q}{2} = g_R \quad (9)$$

Then, the coefficients of the fictitious convection-diffusion equation are simply those of the two independent sub-problems (original and exterior).

**Remark 4** *With the 'no exterior control' choice for the transfer coefficients, we can impose an immersed Robin B.C. leaving independent the solutions of the two domains  $\tilde{\Omega}$  and  $\Omega_e$ : this method is actually an immersed interface method.*

Numerically speaking, this fictitious domain model with immersed transmission conditions requires the discretization scheme to allow jumps of fluxes and solution. A special cell-centered FV scheme has been developed for this purpose in [51] for diffusion problems and extended in [3] for convection-diffusion problems. This cell-centered scheme with jumps at cell boundaries uses the standard  $(2d+1)$ -points stencil only. It is as cheap as the standard cell-centered FV scheme without any jump.

On the Cartesian mesh of the fictitious domain, an approximate stair-case interface  $\Sigma_h$  lying on sides of control volumes is thus defined, see Fig.4. The choice of  $\Sigma_h$  is non-unique but must respect  $|\text{meas}(\tilde{\Omega}_h) - \text{meas}(\tilde{\Omega})| = \mathcal{O}(h)$ . The jump conditions are then applied on  $\Sigma_h$  with discrete transfer coefficients deduced from (9). As for the fictitious domain method with spread interface, a geometric correction  $\epsilon_h$  has then to be introduced in order to take into account the local ratio of surface measures between the original immersed

interface and the approximate one, see [3] for more implementation details. The B.C. on  $\partial\Omega_h \cap \Sigma_h$  must be compatible with the required immersed B.C.

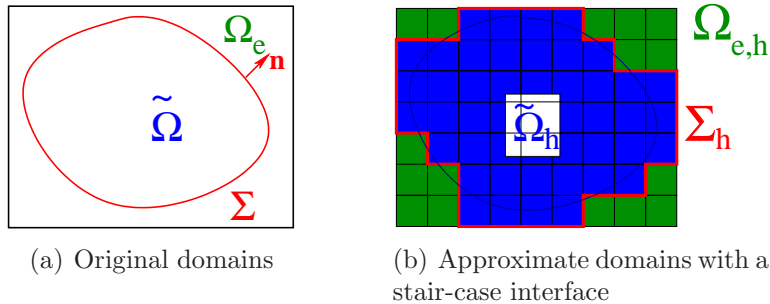


Figure 4: Stair-case interface  $\Sigma_h$  derived from the Cartesian mesh

Academic test cases are proposed in [3]. Among them, one confirms the efficiency of the method to deal with mixed immersed B.C. The theoretical first-order of convergence in space for the  $L^2$ -norm [4] is numerically recovered for all kind of immersed B.C. The modelling error of the surface penalization used for a Dirichlet B.C. seems to be in  $\mathcal{O}(\eta^{1/2})$ .

## 4 Numerical results in an industrial context

We present here an application of the FDMs of Sections 3.1 and 3.2 to industrial engineering computations of the steam generator energy balance equation. Our purpose is to assess these methods, validated on academic numerical test cases [2, 3], in a representative industrial context. Indeed, we want to take advantage of the fictitious domain approach with Cartesian meshes while modifying as less as possible the industrial Genepi code. For the method based on the spread approximation of the immersed interface described in Section 3.1, we directly perform modifications on the FE solver of the Genepi code. As mentioned in Table 1, we only have to add extra terms in (1) to deal with the immersed boundary conditions. We denote this solver FE-ISI. For the method based on a 'thin' approximation of immersed interface supporting immersed jumps, cf. Section 3.2, we replace the genuine FE solver of the Genepi code by a FV solver dedicated to this fictitious domain method with flux and solution jumps. By default, we perform an exterior approximate immersed interface  $\Sigma_h$  (such that  $\tilde{\Omega} \subset \tilde{\Omega}_h$ ) and transfer coefficients verifying the 'no exterior control' choice (9). In this latter case, Genepi only manages the physical correlations and the time evolution. For

efficiency reason, this FV-JEBC solver uses the so-called  $(i, j, k)$  implementation instead of the nodal implementation of the Genepi FE solver. Hence, data-format transformations have to be done between the FV-JEBC solver and Genepi.

Let us precise that the choice of a finite-element discretization is not the only one possible for the ISI method. A finite-volume discretization could also be used with this method. On the other hand, for the JEBC method, a finite-volume discretization is the natural framework.

In the sequel, we aim to compare the FE-ISI solver and the FV-JEBC solver in terms of accuracy and computation efficiency. Moreover a special attention will be paid to the comparison of these FDM solvers with the classical body-fitted FE solver.

#### 4.1 The test case

The industrial test case is build on the CEA half-steam-generator 3-D mock-up Clotaire (at scale 0.7) [52]. We are only interested in the riser simulation. The geometry of the mock-up riser is a 9.17 m high ( $z$  coordinate) and 0.62 m diameter ( $x$  and  $y$  coordinates) half-cylinder filled with a bundle of 184 upside down U-shaped tubes (7.2 m in height) in which flows the hot primary flow. One flow-distribution baffle, nine tube-support plates and one anti-vibrating bar are, respectively located at the bottom, straight and curved part of the tube bundle. Let us notice that, looking at the vertical planar side of this mock-up, the hot leg is on the right side .

The boundary conditions of the secondary-fluid energy balance equation are Dirichlet B.C. at the inflow window (hot leg  $H = 119.3 \text{ KJ.kg}^{-1}$ , cold leg  $H = 118.5 \text{ KJ.kg}^{-1}$ ), see Figure 1, and homogeneous Neumann B.C. elsewhere ( $\nabla H \cdot \mathbf{n} = 0$  with  $\mathbf{n}$  the external normal). The initial condition of the enthalpy is given by a linear profile from  $118.5 \text{ KJ.kg}^{-1}$  on the U-tube plate to  $140 \text{ KJ.kg}^{-1}$  at the outflow.

As no analytic solution is available for this problem, the FE Genepi solution obtained with a 648,388-cell body-fitted-mesh is considered as the reference solution. The mean space steps  $h_x \approx h_y \approx 10^{-2}$  m and  $h_z \approx 2 \cdot 10^{-2}$  m are around the dimensions of the periodic arrangement of the U-tube bundle. In order to be well suited to the physical-domain boundary and the inner structures (U-tube support plates), this unstructured mesh is highly non-uniform.

### 4.1.1 Fictitious-domain meshing and boundary conditions

For the fictitious-domain computations, the half-cylinder geometry of the physical domain  $\tilde{\Omega}$  is immersed into a rectangular parallelepiped  $\Omega$  (cf. figures 5(a) and 5(b)). Three Cartesian meshes of the fictitious domain  $\Omega$  involving 7,200, 57,600 and 460,800 cells are built. The space step is divided by two from one mesh to the next one. The space steps of the finest mesh ( $h_x \approx 1.3 \cdot 10^{-2}$  m,  $h_y \approx 1.4 \cdot 10^{-2}$  m and  $h_z \approx 2.3 \cdot 10^{-2}$  m) are around those of the reference simulation.

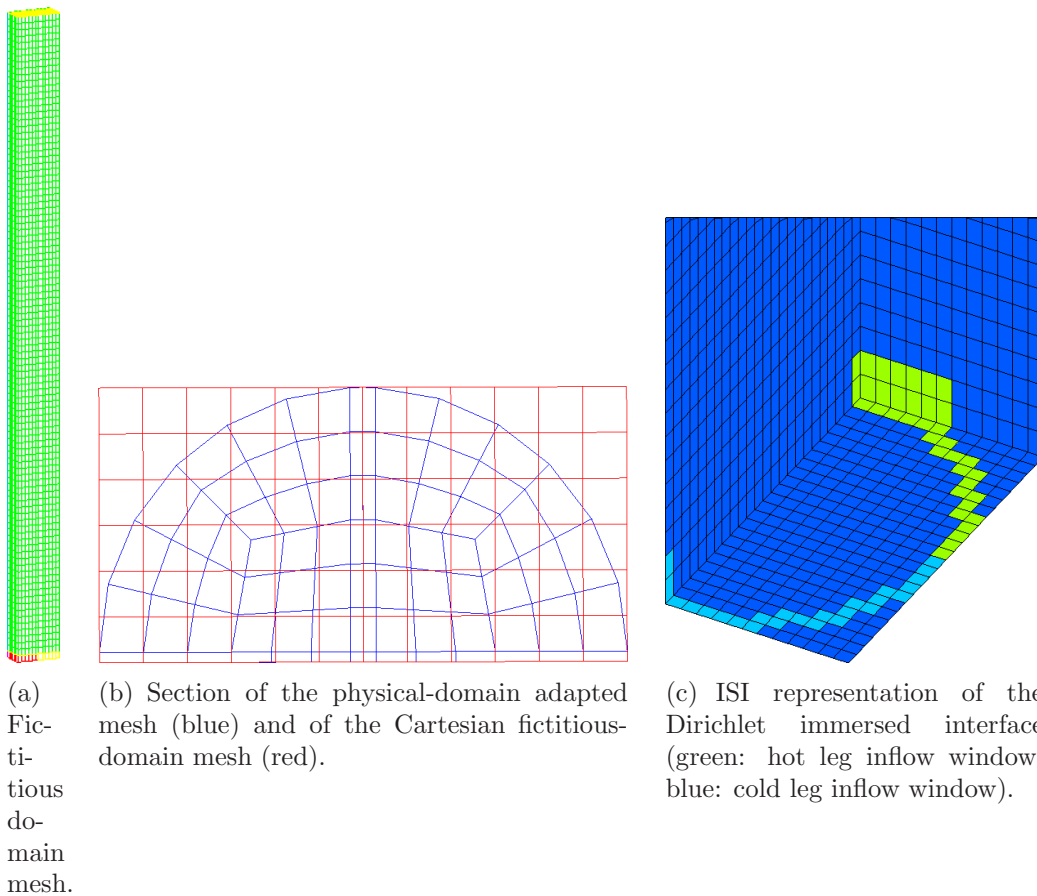


Figure 5: Physical domain adapted mesh, Cartesian fictitious domain mesh and immersed Dirichlet boundary conditions for the spread approximation of the interface.

The geometry of the approximate immersed interface is computed from the intersection between the body-fitted mesh and the fictitious Cartesian mesh, see Figure 5(b). For each cell cut by the interface, we dispose of the interface

measure inside the cell. About 15% of the Cartesian cells are located in the external domain that is considered as an *obstacle* with a small porosity for the fluid.

The Robin immersed boundary condition formalism (cf. section 3) allows us to deal with the mixed immersed boundary conditions (Dirichlet/Neumann) under study. By this way, we test our modeling of the immersed Robin B.C. with extreme values of the coefficient  $\alpha_R$ . Concerning the own B.C. of the fictitious domain ( $\partial\Omega \setminus \partial\tilde{\Omega}$ ), we also impose mixed boundary conditions: homogeneous Neumann B.C. everywhere except on faces in front of the immersed Dirichlet boundaries. For these particular faces, we impose the Dirichlet value of the corresponding immersed boundary.

#### 4.1.2 Numerical features

It is worth to give some precisions concerning the Genepi code and the fictitious domain solvers. In Genepi, the enthalpy ( $H$ ), the porosity ( $\beta$ ) as well as the drift term are expressed in a  $\mathcal{Q}_1$  finite element base, while the density ( $\rho$ ), the pressure ( $P$ ), the advection mass flux ( $\mathbf{G}$ ), the turbulent diffusion coefficient ( $\chi_T$ ) and the heat source term ( $Q$ ) are expressed in a  $\mathcal{Q}_0$  finite element base. Whatever the solver is, the primary-fluid energy balance equation is solved by 1-D FE computations coupled at each time step with the secondary-fluid energy balance equation through the source term  $Q$  of (1). For the secondary-fluid equation, the mixture pressure  $P$  and mass flux  $\mathbf{G}$  fields are interpolated on cell barycenters from a complete N-S resolution by, respectively, a constant and a trilinear interpolation, see Figure 6(a). The stationary regime is found by a semi-implicit march-in-time algorithm, in which the time step  $\Delta t$  is limited to 2.5 times the CFL time step, and is defined by the following criteria  $\|\xi^{n+1} - \xi^n\|_{L^2} / \|\xi^n\|_{L^2} \leq 10^{-5} \Delta t$  with  $\xi$  represents the unknowns.

The characteristics of the FE-ISI solver are those of the Genepi code. Hence the FE-ISI discretization algorithm is second-order in space for the diffusion and the convection terms and second-order in time (Crank-Nicholson scheme). Balancing tensor-diffusivity correction [53] is used to prevent spurious oscillations introduced by the central-difference discretization of the convection term. The FV-JEBC solver is based on constant cell values. Then cell projections of  $\mathcal{Q}_1$  FE quantities are done between Genepi and this solver. In the FV-JEBC solver, the discretization of the diffusion term is also second-order in space, but the convection scheme is only first-order in space (upwind scheme). The time scheme is the first-order semi-implicit Euler scheme. At each time step, linear systems are smoothed by some ILLU [54] (Incomplete Line LU factorization)-preconditioned CGS [55] iterations (FE-ISI) or



diagonal-preconditioned Bi-CGStab [56] iterations (FV-JEBC). Typically, a maximum number of 20 iterations is done. These iterations are early stopped if the initial residual is reduced by a factor  $10^7$ .

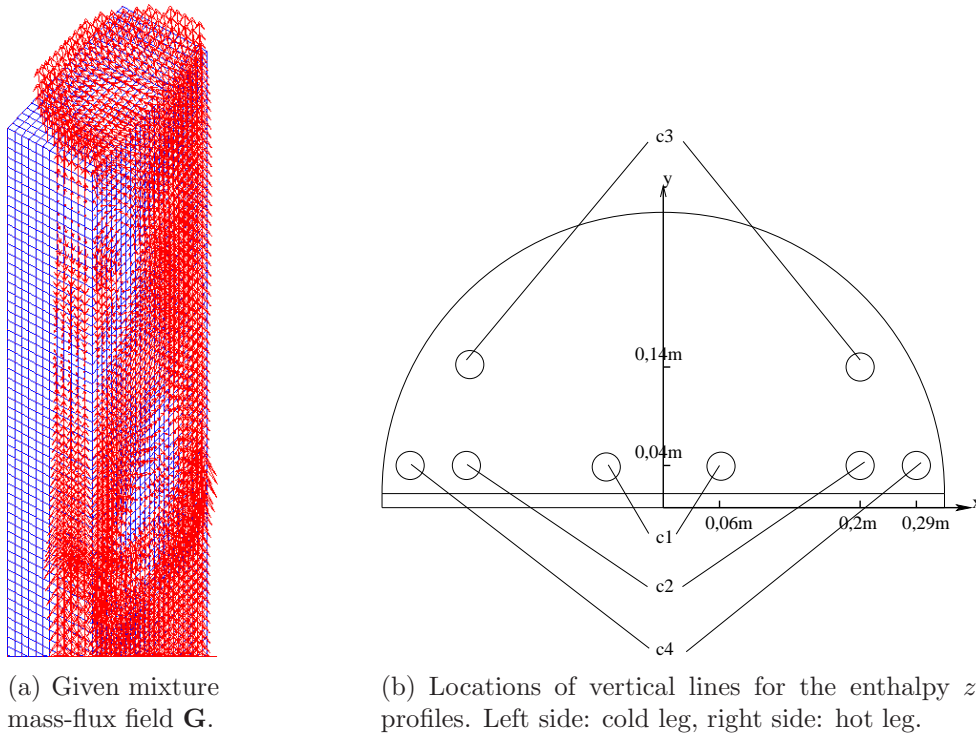


Figure 6: Mixture mass-flux field and locations of enthalpy  $z$ -profiles.

## 4.2 Results

Results are appreciated on criteria concerning the global primary/secondary-fluid heat exchange  $W_{ech}$ , the secondary-fluid enthalpy or the computation cost. Comparisons are made between the fictitious domain solutions and the reference solution. For example,  $L^2$ -norm errors of the enthalpy are performed in the whole physical domain as well as along several vertical lines (enthalpy  $z$ -profiles), see Figure 6(b).

The stationary regime is reached after respectively 441, 1,080 and 2,719 time steps for the 7,200, 57,600 and 460,800-cell meshes whatever the FD solver is. This is similar to the number of time steps required for body-fitted simulations with the same stationary-regime criteria and a similar minimal space

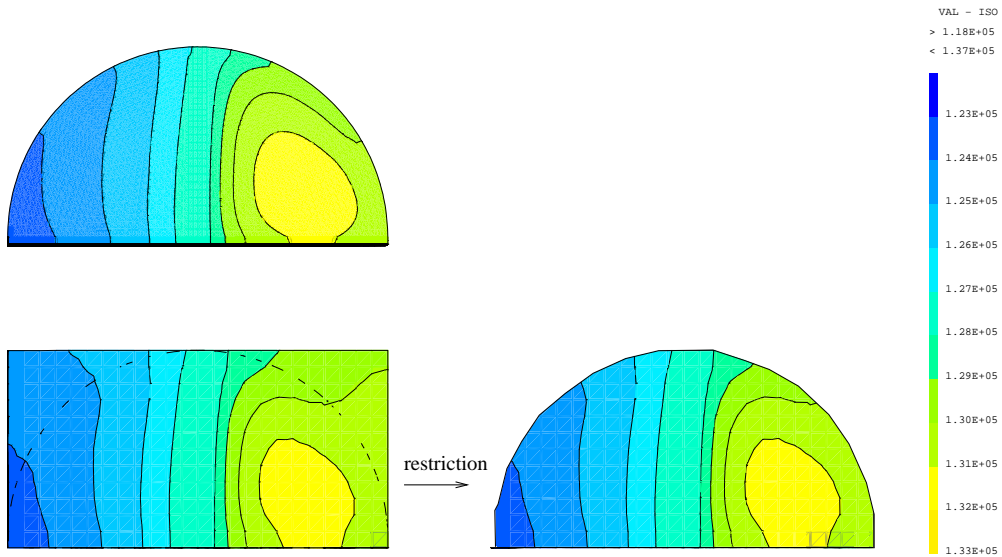
step.

Following the space step  $h$ , the global primary/secondary-fluid heat exchange is recovered with a relative error around  $10^{-3}$  to  $10^{-5}$  in comparison with the reference computation, see Table 2. The heat exchange  $W_{ech}$  quickly converges toward the reference value. For the finest mesh, we seem to reach the industrial computational precision. Hence, both fictitious domain methods have the same level of accuracy as regard to this global quantity that is of interest for industrial steam-generator simulations.

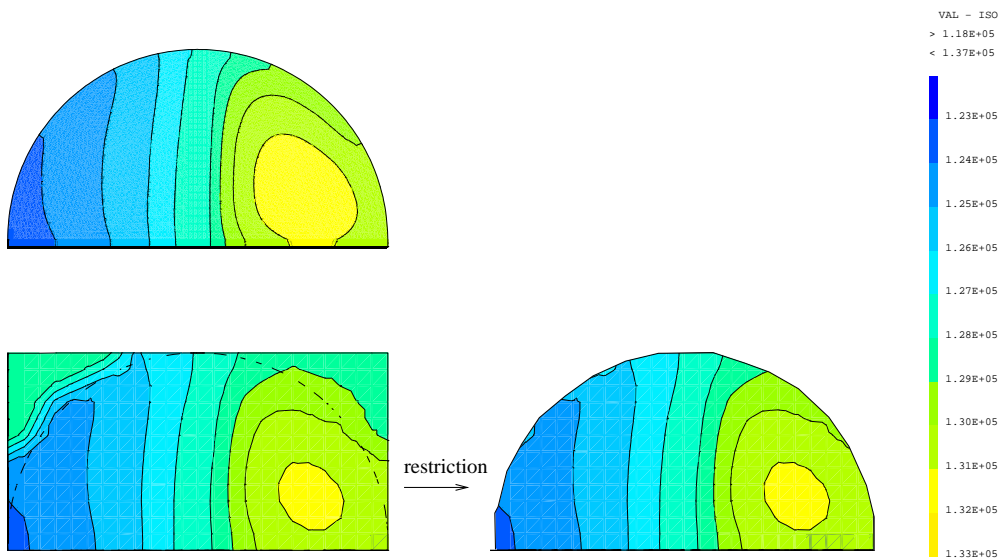
Cell #	7,200	57,600	460,800
FE-ISI	$2.4 \cdot 10^{-3}$	$5.9 \cdot 10^{-4}$	$8.0 \cdot 10^{-5}$
FV-JEBC	$1.8 \cdot 10^{-3}$	$7.2 \cdot 10^{-4}$	$7.0 \cdot 10^{-6}$

Table 2: Absolute values of the relative error on primary/secondary-fluid heat exchange  $W_{ech}$ . The reference heat exchange is equal to  $1.48 \cdot 10^6$  W.

Figure 7(a) (resp. 7(b)) shows the enthalpy isovalues obtained on a horizontal cut plane at half elevation of the SG mock-up with the ISI method (resp. JEBC method). In these Figures, the reference solution (top) is compared with FE-ISI and FV-JEBC solutions obtained with the 57,600-cell mesh (bottom). Qualitatively the results are very close, particularly for the FE-ISI method. Indeed, the error associated to the FV-JEBC method seems to be higher than the error obtained for the FE-ISI method. A first conclusion drawn from these simulations is that the immersed boundary method is slightly more accurate than the immersed interface method.



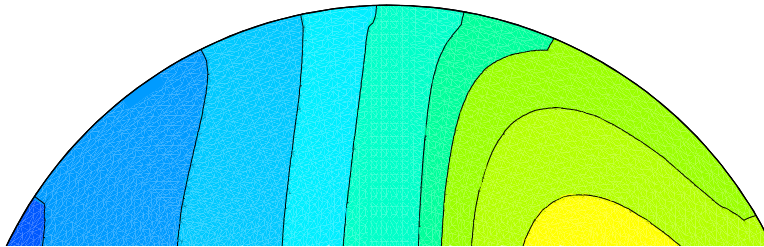
(a) FE-ISI solution.



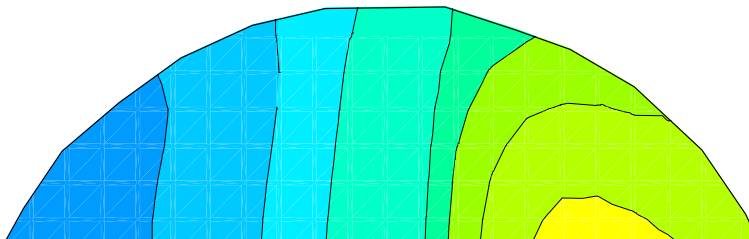
(b) FV-JEBC solution.

Figure 7: Enthalpy isovalues for a horizontal cut plane at 5 m (half elevation). For each figure, top: body-fitted mesh reference solution; bottom: 57,600-cell Cartesian mesh FD solution.

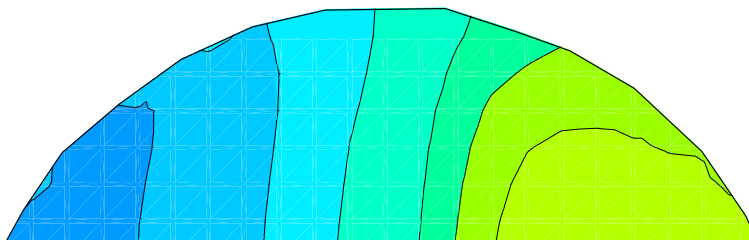
We can notice that, as expected, the isovalue lines of all simulations are perpendicular to the wall (homogeneous Neumann boundary condition), see Figures 8(a), 8(b) and 8(c). On the zoom of the FV-JEBC solution (Figure 8(c)), this feature is sometimes locally masked by the bad post-processing of the strong varying solution near the immersed boundary.



(a) Body-fitted mesh reference solution.



(b) FE-ISI solution (57,600-cell Cartesian mesh).



(c) FV-JEBC solution (57,600-cell Cartesian mesh).

Figure 8: Zoom near the wall (immersed boundary) of the physical domain enthalpy isovalues. Horizontal cut plane at 5 m (half elevation).

Figures 7(b) and 9 confirm that for the JEBC method with 'no exterior control' transfer coefficients, the exterior domain has no influence on the physical domain solution. Moreover, we can see on Figure 9 that this method definitely allows (at least) jumps of solution!

The  $L^2$ -norm error over the whole physical domain of the secondary-fluid

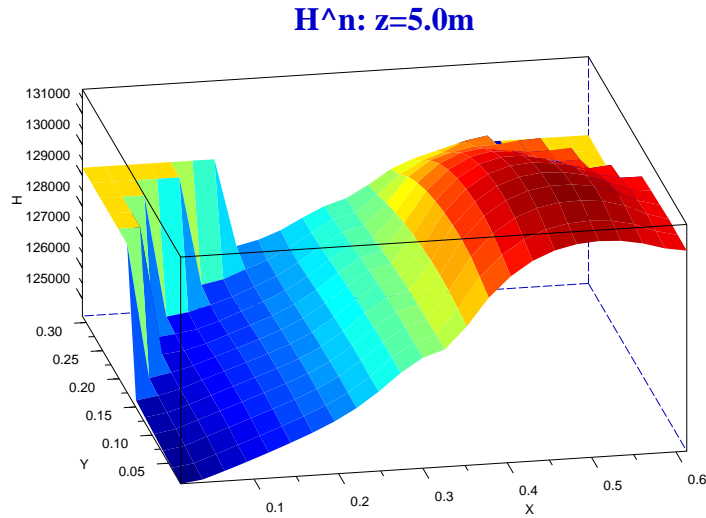


Figure 9: 3D representation of the enthalpy isovalues for a horizontal cut plane at 5 m (half elevation) of the 57,600-cell Cartesian mesh FV-JEBC solution.

enthalpy is reported in Table 3. Also, we indicate the  $L^2$ -norm error for the part of the physical domain in regard to the inflow window Dirichlet immersed B.C.. As a whole, the space convergence order of the FDMs in the  $L^2$ -norm tends to  $\mathcal{O}(h)$  before reaching the discretization error of the reference solution. This is coherent with the academic test cases [2, 3] and the convergence analysis [4] that conclude to first-order methods. Richardson's extrapolation formula [57] enables us to obtain the intrinsic convergence order of the enthalpy  $L^2$ -norm : we obtain a convergence in  $\mathcal{O}(h^{1.23})$  for the FE-ISI solution and in  $\mathcal{O}(h^{1.15})$  for the FV-JEBC solution over the whole physical domain. This lower-bound of the  $L^2$ -norm of the error is coherent with the expected first-order convergence of the methods.

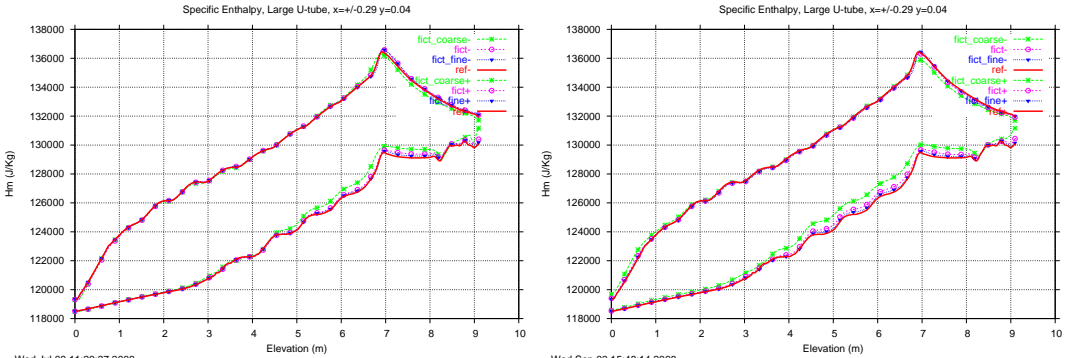
Then the first-order discretization in space (convection) of the FV-JEBC method is optimal since using higher-order discretization schemes does not improve the overall accuracy. These results also confirm that the ISI method is globally more accurate than the JEBC method with 'no exterior control' coefficients. However, it is worth to underline that the JEBC method leads to the lower error values for the subdomain in front of the inflow window and then seems to better simulate Dirichlet B.C..

The  $z$ -profiles of the secondary-fluid enthalpy are plotted in Figures 10(a) and 10(b) for the  $c4$  lines (see Figure 6(b)). These figures enable us to conclude

Cell #	Whole physical domain			Inflow window part		
	7,200	57,600	460,800	7,200	57,600	460,800
	(convergence order)			(convergence order)		
FE-ISI	$3.43 \cdot 10^{-3}$	$1.67 \cdot 10^{-3}$	$1.08 \cdot 10^{-3}$	$2.81 \cdot 10^{-3}$	$2.08 \cdot 10^{-3}$	$1.06 \cdot 10^{-3}$
	(-)	(1.03)	(0.63)	(-)	(0.43)	(0.95)
FV-JEBC	$4.09 \cdot 10^{-3}$	$2.14 \cdot 10^{-3}$	$1.36 \cdot 10^{-3}$	$1.82 \cdot 10^{-3}$	$7.76 \cdot 10^{-4}$	$9.92 \cdot 10^{-4}$
	(-)	(0.93)	(0.65)	(-)	(1.23)	(-0.36)

Table 3: Relative  $L^2$ -norm error of the secondary-fluid enthalpy over the whole physical domain or the part of this domain in front of the inflow window.

to the good accuracy of both FDMs. Moreover the mesh convergence is recovered. We can notice that, for the coarsest mesh, the ISI profiles are closer to the reference profiles than the JEBC ones.

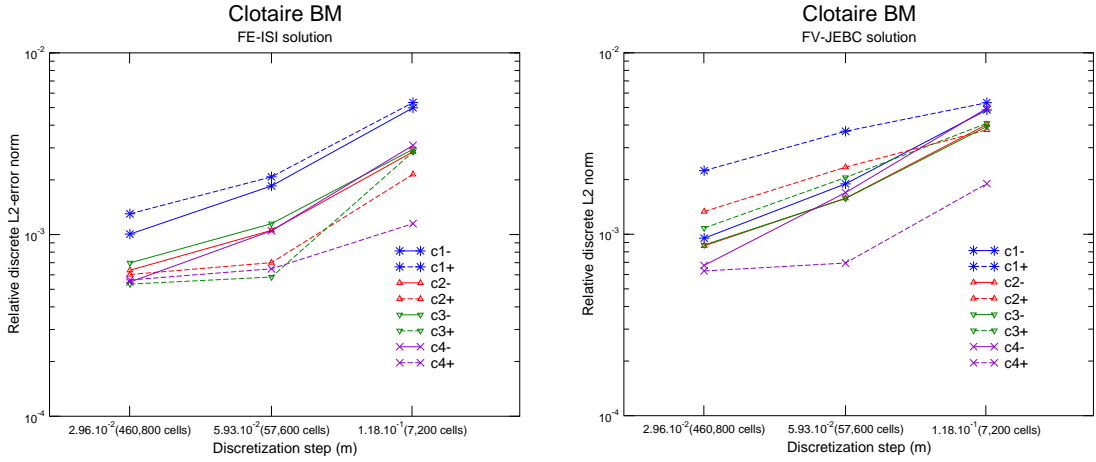


(a) Immersed spread interface method (FE-ISI). (b) Thin interface with immersed jumps method (FV-JEBC).

Figure 10: Secondary-fluid enthalpy profiles for the  $c4$  lines. The notation '-' for the lower curves (respectively '+' for the upper curves) is related to the cold leg (respectively the hot leg). The legends 'fict\_coarse', 'fict' and 'fict\_fine' refer to the Cartesian mesh sizes (respectively 7,200; 57,600 and 460,800 cells).

The relative  $L^2$ -norm errors for the set of enthalpy  $z$ -profiles are plotted in Figures 11(a) and 11(b). As for the results from Table 3, these errors are always lower than  $5 \cdot 10^{-3}$ . These Figures confirm the previous qualitatively results (cf. Figures 10(a) and 10(b)): both FDMs converge with the mesh size. For some profiles (e.g. line  $c4+$ ), the fictitious domain solution early reaches the reference solution. Again, the  $L^2$ -norm error globally decreases with order  $\mathcal{O}(h)$  in space, see Table 4. In view of these representations,

the previous conclusion is still valid: the FE-ISI solver is a little bit more accurate than the FV-JEBC solver. However this latter solver converges in a more monotone way.



(a) Immersed spread interface method (FE-ISI).

(b) Thin interface with immersed jumps method (FV-JEBC).

Figure 11: Relative  $L^2$ -norm error versus the space step  $h$  for the enthalpy profiles ( $0 \leq z \leq 8$  m). The notation '-' (respectively '+') is related to the cold leg (respectively the hot leg).

	FE-ISI			FV-JEBC		
	Ratio $r$		Aver. order	Ratio $r$		Aver. order
C1-	2.68	1.84	1.14	2.56	1.99	1.14
C1+	2.56	1.60	1.04	1.43	1.65	0.77
C2-	2.71	1.66	1.09	2.55	1.83	1.09
C2+	3.06	1.16	1.06	1.61	1.76	0.84
C3-	2.57	1.64	1.06	2.49	1.81	1.07
C3+	4.89	1.09	1.50	1.99	1.90	0.97
C4-	2.95	1.91	1.22	2.92	2.52	1.36
C4+	1.77	1.15	0.73	2.74	1.11	0.96

Table 4: Arithmetic-averaged orders of convergence in space of the  $L^2$ -norm error of the enthalpy  $z$ -profiles. Ratio  $r$  between the  $z$ -profile  $L_2$ -norm errors computed using one grid and the next finer one. Convergence order defined as  $\ln(r)/\ln(2)$ .

Now, we focus on the computational efficiency point of view. It is more

linked to the space discretization and to the data structures than to the FDMs themselves. A first difference between the solution procedures concerns the minimal value of the  $\eta$  coefficient that can be effectively performed. In case of the FV-JEBC solver, this value can be as small as  $10^{-14}$  without any problem to solve the linear systems even using simple preconditioner as the diagonal one. On the other hand, for the FE-ISI solver, the value of  $\eta$  is bounded by  $10^{-6}$  (it can be decreased to  $10^{-9}$ , but the algorithm convergence is slower). This is due to the fact that the mass matrix resulting from the  $Q_1$  FE discretization of the reaction term involves off-diagonal entries with large values. The built of a good preconditioner is then by far not trivial. No off-diagonal terms appear with the cell-centered FV scheme. Hence a diagonal preconditioner is sufficient to improve the matrix condition number.

Cell #	FE-ISI			FV-JEBC			Ref. 648,388
	7,200	57,600	460,800	7,200	57,600	460,800	
Trans.	-	-	-	$1.30 \cdot 10^{-2}$	$1.65 \cdot 10^{-1}$	1.35	-
Ass.	$> 1.35 \cdot 10^{-1}$	$> 1.10$	$> 7.93$	$3.70 \cdot 10^{-2}$	$3.67 \cdot 10^{-1}$	3.05	unknown
Sol.	$5.70 \cdot 10^{-2}$	$5.59 \cdot 10^{-1}$	4.23	$2.80 \cdot 10^{-2}$	$2.06 \cdot 10^{-1}$	5.41	unknown
$\Delta t$	$4.46 \cdot 10^{-1}$	3.50	25.90	$3.27 \cdot 10^{-1}$	4.64	24.02	37.60
Mem.	67	240	1,588	65	230	1,554	2,148

Table 5: Comparison of the computational efficiency between the FD simulations and the reference one. CPU time are expressed in seconds and memory in MBytes. Linear system solvers: ILLU-preconditioned CGS for the FE-ISI method and the reference simulation; diagonal-preconditioned Bi-CGStab for the FV-JEBC method. 'Trans.': data translation CPU time; 'Ass.': matrix assemblage CPU time; 'Sol.': linear system solver CPU time; ' $\Delta t$ ': a time step CPU time; 'Mem.': memory.

Table 5 shows the CPU time used by the simulations. The data translation time is the extra CPU time required to convert the FE mesh-oriented data storage (node/face/cell connexions) into table-oriented data storage ( $((i, j, k)$  indexes) and *vice-versa*. The table-oriented format of the data storage and the smaller number of matrix entries give advantage to the FV-JEBC solver: the matrix assemblage CPU time is the smallest one. The FV-JEBC method also offers better performances for the linear system solver: lower matrix band size (cache memory effect for small numbers of unknowns) and easier preconditioning. As a whole, taking into account the data translation time, the FV-JEBC solver requires only as CPU time as the FE-ISI solver to compute a time step. The analysis of the memory leads to the same conclusion



since the FV-JEBC memory shown in Table 5 includes extra memory for the data translation. Then, the performances concerning data storage and CPU time of the FV-JEBC method should be even more obvious with a code fully exploiting the  $(i, j, k)$  index format.

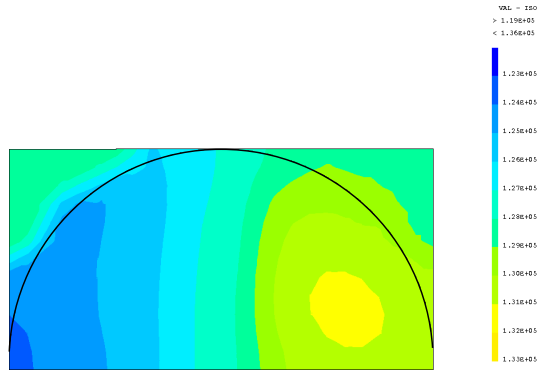
**Remark 5** *Other transfer coefficient choices in (8) have been tested for the FV-JEBC method leading to similar numerical results than the 'no exterior control' choice (9). For example, if we want to preserve the continuity of the solution ( $u_{\Sigma}^+ \approx u_{\Sigma}^- := u_{\Sigma}$ ), we have to set  $\alpha = \alpha_R$ ,  $\beta = 1/\eta$ ,  $q = -g_R$  and  $g = 0$ . If we look at the 57,600-cell mesh solutions (cf. Figures 12(a) and 12(b)), we can remark that the numerical results are very close. Actually the only differences can be found in the increase of smoothness near the immersed interface brought by the 'solution continuity' choice (no enthalpy jump) and in the easier solve of the linear systems. But the main drawback is the necessity to control the external part of the fictitious domain setting  $-(\mathbf{a} \cdot \nabla u) \cdot \mathbf{n}|_{\Sigma}^{\pm} \approx 0$  through  $\mathbf{a}_{\Omega_{e,h}} = \eta \mathbf{Id}$  as in the spread interface approach.*

**Remark 6** *Likewise, we have tested other definitions of the approximate immersed-interface geometry for the JEBC method. In particular, defining the approximate immersed interface as the set of faces cut by the exact immersed interface brings a small increase of accuracy for the coarsest Cartesian mesh simulation (several percents only).*

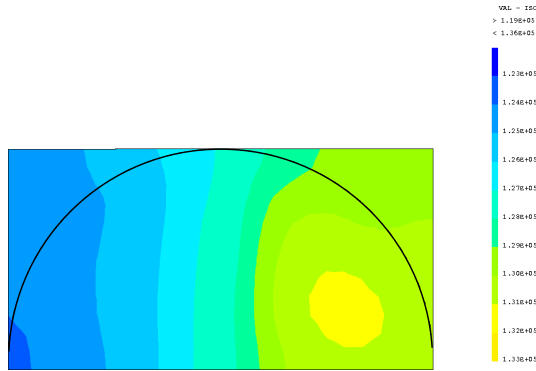
## 5 Conclusion

In this paper we have tested and compared two fictitious domain methods enforcing immersed boundary conditions. The first one is based on a spread approximation of the immersed interface [2], while the second one is based on immersed jumps supported by a thin approximation of the immersed interface [3]. Both methods can deal with mixed immersed boundary conditions (Dirichlet, Neumann and/or Robin B.C.). Actually, the fictitious domain method with immersed jumps can be used as an immersed interface method in the sense where the external fictitious domain is related to the physical domain by transmission conditions. This work has been devoted to assess these methods, already validated on academic test cases in [2, 3], in a representative industrial test case: the thermal-hydraulic simulation of nuclear power plant components. As a first inside in this task, we have considered the computation of the energy balance-equation of the steam-generator mock-up Clotaire. We have paid a particular attention to the easiness of the introduction of this kind of approach in existing industrial codes.

For this purpose, the immersed spread interface method has been directly



(a) 'No exterior control' choice.



(b) 'Solution continuity' choice.

Figure 12: Effect of the transfer coefficients on the enthalpy field (horizontal cut plane at 5 m; half elevation) for the FV-JEBC solver with a 57,600-cell Cartesian mesh.

introduced in the finite-element Genepi code (FE-ISI solver), adding extra terms in the discretized equations. On the other hand, the native Genepi energy-balance equation solver can be also replaced by a finite-volume fictitious-domain solver (FV-JEBC solver) to test the method with immersed jumps. Hence, these two methods present good properties concerning their capabilities to be easily introduced in an pre-existing industrial code. By this way, we have tested the generic immersed B.C. modeling, based on the Robin formalism, to deal with an industrial problem with immersed mixed Dirichlet/Neumann B.C..

The reported simulations illustrate the really satisfactory performances of these fictitious domain methods in a industrial context. Although they use different modelling to handle immersed B.C. as well as different space and

time discretizations, both fictitious domain methods lead to the same accuracy as a whole: the relative  $L^2$ -norm errors calculated from a reference solution quickly decrease under 0.1%. The theoretical first-order mesh convergence is recovered [4]. For comparable space steps, the fictitious domain methods enable us to save about 30% of the global reference computational cost involving CPU time and memory. This gain should be even more appreciable with a code fully exploiting the  $(i, j, k)$  data-storage format, well suited for fictitious domain simulations on Cartesian meshes. Generally speaking, the FE-ISI errors are slightly lower than the FV-JEBC ones. In counterpart, the FV-JEBC solver already benefits of the optimal data storage and offers the great possibility of obtaining the physical domain solution in a totally uncoupled way with the external domain modeling. All these conclusions give us confidence to extend these fictitious domain methods to the Navier-Stokes equations for fully simulate nuclear components and study fluid-structure interactions.

Acknowledgement :This work has been achieved in the framework of the NEPTUNE project, financially supported by CEA (Commissariat à l’Energie Atomique), EDF (Electricité de France), IRSN (Institut de Radioprotection et de Sûreté Nucléaire) and AREVA-NP.

## References

- [1] I. Ramière, Ph. Angot, and M. Belliard. Fictitious domain methods to solve convection-diffusion problems with general boundary conditions. In *Proc. in the 17th Computational Fluid Dynamics Conference AIAA*, Toronto, Canada, June 2005.
- [2] I. Ramière, Ph. Angot, and M. Belliard. A fictitious domain approach with spread interface for elliptic problems with general boundary conditions. *Computer Methods in Applied Mechanics and Engineering*, 196(4-6):766–781, 2007.
- [3] I. Ramière, Ph. Angot, and M. Belliard. A general fictitious domain method with immersed jumps and non-conforming structured mesh. *Journal of Computational Physics*, 225(2):1347–1387, 2007.
- [4] I. Ramière. Convergence analysis of the Q1-finite element method for elliptic problems with non-boundary-fitted meshes. *International Journal for Numerical Methods in Engineering*, 75(9):1007–1052, 2008.
- [5] A. Guelfi, D. Bestion, M. Boucker, P. Boudier, P. Fillion, M. Grandotto, J.-M. Hérard, E. Hervieu, and P. Péturaud. NEPTUNE - A New Soft-

- ware Platform for Advanced Nuclear Thermal Hydraulics. *Nuclear Science and Engineering*, 156:281–324, 2007.
- [6] S. Clerc. Numerical simulation of the Homogeneous Equilibrium Model for Two-phase Flows. *Journal of Computational Physics*, 161(1):354–375, 2000.
- [7] M. Grandotto and P. Obry. Calculs des écoulements diphasiques dans les échangeurs par une méthode aux éléments finis. *Revue Européenne des Eléments Finis*, 5(1):53–74, 1996.
- [8] I. Toumi, A. Bergeron, D. Gallo, E. Royer, and D. Caruge. Flica-4: a three-dimensional two-phase flow computer code with advanced numerical methods for nuclear applications. *Nuclear Engineering and Design*, 200(1-2):139–155, 2000.
- [9] H. Schlichting. *Boundary Layer Theory*. Mac Graw Hill, New York, USA, 1968.
- [10] N. Zuber and J.A. Findlay. Average Volumetric Concentration in Two-Phase Flow Systems. *J. Heat Transfert*, 87(4):453–468, 1965.
- [11] G. S. Lellouche and B. A. Zolotar. Mechanistic Model For Predicting Two-Phase Void Fraction For Water in Vertical Tubes, Channels, and Rod Bundles. Technical report, EPRI, 1982.
- [12] Ph. Angot, J. P. Caltagirone, and K. Khadra. A comparison of locally adaptive multigrid methods: L.D.C., F.A.C. and F.I.C. In *NASA Conf. Publ. CP-3224*, volume 1, pages 275–292, 1993.
- [13] O. Roussel and K. Schneider. An adaptive multiresolution method for combustion problems : Application to flame ball-vortex interaction. *Computers and Fluids*, 34:817–831, 2005.
- [14] K. Khadra, Ph. Angot, S. Parneix, and J.P. Caltagirone. Fictitious Domain Approach for Numerical Modelling of Navier-Stokes Equations. *International Journal for Numerical Methods in Fluids*, 34(8):651–684, 2000.
- [15] M.A. Hyman. Non-iterative numerical solution of boundary-value problems. *Applied Scientific Research, Section B*, 2(1):325–351, 1952.
- [16] V.K. Saul’ev. On the solution of some boundary value problems on high performance computers by fictitious domain method. *Siberian Math. Journal*, 4(4):912–925, 1963 (in Russian).

- [17] L.A. Rukhovets. A remark on the method of fictive domains. *Differential Equations*, 3(4):114–121, 1967 (in Russian).
- [18] V.D. Kopčenov. A method of fictitious domains for the second and third value problems. *Trudy Mat. Inst. Steklov*, 131:119–127, 1974 (in Russian).
- [19] G. P. Astrakhantsev. Method of fictitious domains for a second-order elliptic equation with natural boundary conditions. *USSR Computational Mathematics and Mathematical Physics*, 18:114–121, 1978.
- [20] G.I. Marchuk. *Methods of Numerical Mathematics*, volume 2 of *Application of Math*. Springer-Verlag, New York, USA, 1982 (1st ed. 1975).
- [21] A. Parvizian, A. Düster, and E. Rank. Finite cell method - h- and p-extension for embedded domain problems in solid mechanics. *Comput. Mech.*, 41(1):121–133, 2007.
- [22] R. Cortez and M. Minion. The blob projection method for immersed boundary problems. *Journ. of Comp. Physics*, 161(2):428–453, 2000.
- [23] A. Wiegman and K.P. Bube. The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions. *SIAM J. Numer. Anal.*, 37(3):827–862, 2000.
- [24] A.L. Fogelson and J.P. Keener. Immersed interface methods for Neumann and related problems in two and three dimensions. *SIAM J. Sci. Comput.*, Vol 22, pp. 1630-1684, 2000.
- [25] F. Gibou, R. P. Fedkiw, L.-T. Cheng, and M. Kang. A Second-Order-Accurate Symmetric Discretization of the Poisson Equation on Irregular Domains. *Journal of Computational Physics*, 176(1):205 – 227, 2002.
- [26] C. S. Peskin. Flow patterns around heart valves: A numerical method. *J. Comput. Phys.*, 10(2):252–271, 1972.
- [27] C. S. Peskin. The immersed boundary method. *Acta Numerica*, 11(3):479–517, 2002.
- [28] P. McCorquodale, Ph. Colella, and H. Johansen. A Cartesian Grid Embedded Boundary Method for the Heat Equation on Irregular Domains. *J. Comput. Phys.*, 173(2):620–635, 2001.

- [29] T. Ye, R. Mittal, H.S. Udaykumar, and W. Shyy. An Accurate Cartesian Grid Method for Viscous Incompressible Flows with Complex Immersed Boundaries. *J. Comput. Phys.*, 156:209–240, 1999.
- [30] E. A. Fadlun, R. Verzicco, P. Orlandi, and J. Mohd-Yusof. Combined Immersed-Boundary Finite-Difference Methods for Three-Dimensional Complex Flow Simulations. *Journal of Computational Physics*, 161:35 – 60, 2000.
- [31] T. Ikeno and T. Kajishima. Finite-difference immersed boundary method consistent with wall conditions for incompressible turbulent flow simulations. *Journal of Computational Physics*, 226:1485 – 1508, 2007.
- [32] R. Glowinski, T. W. Pan, and J. Peraux. A fictitious domain method for dirichlet problem and applications. *Computer Methods in Applied Mechanics and Engeneering*, 111(3-4):283–303, 1994.
- [33] F. Collino, P. Joly, and F. Millot. Fictitious domain method for unsteady problems: application to electromagnetic scattering. *J. Comput. Phys.*, 138(2):907–938, 1997.
- [34] R. Glowinski and Y. Kuznetsov. Distributed lagrange multipliers based on fictitious domain method for second order elliptic problems. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1498–1506, 2007.
- [35] Ph. Angot, Ch.-H. Bruneau, and P. Fabrie. A penalization method to take into account obstacles in incompressible viscous flows. *Numerische Mathematik*, 81(4):497–520, 1999.
- [36] A. Sarthou, S. Vincent, J.-P. Caltagirone, and Ph. Angot. Eulerian-lagrangian grid coupling and penalty methods for the simulation of multiphase flows interacting with complex objects. *International Journal for Numerical Methods in Fluids*, 56(8):1093–1099, 2008.
- [37] Y. H. Tseng and J. H. Ferziger. A ghost-cell immersed boundary method for flow in complex geometry. *Journal of Computational Physics*, 192(2):593 – 623, 2003.
- [38] B. Maury. A Fat Boundary Method for the Poisson problem in a domain with holes. *SIAM, J. of Sci. Comput.*, 16(3):319–339, 2001.
- [39] S. Bertoluzza, M. Ismail, and B. Maury. Analysis of the fully discrete fat boundary method. *accepted in Numerische Mathematik*, 2010.

- [40] X. Li, J. Lowengrub, A. Rätz, and A. Voigt. Solving PDEs in complex geometries: A diffuse domain approach. *Commun. Math. Sci.*, 7(1):81 – 107, 2009.
- [41] Z. Jomaa and C. Macaskill. The Shortley Weller embedded finite-difference method for the 3D Poisson equation with mixed boundary conditions. *J. Comput. Phys.*, 229(10):3675–3690, 2010.
- [42] G.J. Wagner, N. Moës, W.K. Liu, and T. Belytschko. The extended finite element method for rigid particles in stokes flow. *Int. journ. for numerical methods in engineering*, 51(3):293–313, 2001.
- [43] J. Haslinger and Y. Renard. A new fictitious domain approach inspired by the extended finite element method. *SIAM J. Numer. Anal.*, 47(2):147–1499, 2009.
- [44] R. J. LeVeque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.*, 31:1019–1044, 1994.
- [45] Z. Li. An overview of the immersed interface method and its applications. *Taiwanese Journal of Mathematics*, 7(1):1–49, 2003.
- [46] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher. A non-oscillatory eulerian approach to interfaces in multi-material flows (the ghost fluid method). *Journal of Computational Physics*, 152(2):457 – 492, 1999.
- [47] Y. C. Zhou and G. W. Wei. On the fictitious-domain and interpolation formulations of the matched interface and boundary (MIB) method. *J. Comput. Phys.*, 219(1):228–246, 2006.
- [48] A. Sarthou, S. Vincent, Ph. Angot, and J.-P. Caltagirone. The algebraic immersed interface and boundary method for the elliptic equations with discontinuous coefficients. *Submitted*, 2010.
- [49] Ph. Angot. A unified fictitious domain model for general embedded boundary conditions. *C. R. Acad. Sci. Paris, Sér. I*, 341(11):683–688, 2005.
- [50] H. Brezis. *Analyse fonctionnelle, Théorie et applications*. Dunod, 2000.
- [51] Ph. Angot. A model of fracture for elliptic problems with flux and solution jumps. *C.R. Acad. Sci. Paris, Ser I Math.*, 337 (6) 425-430, 2003.

- [52] J.L. Campan and J.C. Bouchter. Steam generator experiment for advanced computer code qualification : CLOTAIRE international program. In *Third International Topical Meeting on Nuclear Power Plant Thermohydraulics and Operations*, Seoul, April 1988.
- [53] P.M. Gresho, S.T. Chan, R. L. Lee, and C.D Upson. A modified Finite Element Method for Solving the Time-Dependent Incompressible Navier Stokes Equations (Part 1: Theory). *Int. J. Num. Methods in Fluids*, 4(6):557–598, 1984.
- [54] J.A. Meijerink and H.A. Van Der Vorst. Guidelines for the usage of incomplete decompositions in solving linear equations as they occur in practical problems. *Journal of Computational Physics*, 44(1):134–155, 1981.
- [55] P. Sonneveld. CGS, a fast Lanczos-type solver for nonsymmetric linear system. *SIAM J.Sci. Statist.Comput.*10, 36-52, 1989.
- [56] H.A. Van Der Vorst. BI-CGStab: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.*, 13(2):631–644, 1992.
- [57] L. F. Richardson. The approximate arithmetical solution by finite differences of physical problems including differential equations, with an application to the stresses in a masonry dam. *Philosophical Transactions of the Royal Society of London, Series A*, 210:307–357, 1911.