



# **Fast 3D computations of compressible flow discharge in buildings and complex networks**

Alexandre Chiapolino, François Fraysse, Richard Saurel

## **► To cite this version:**

Alexandre Chiapolino, François Fraysse, Richard Saurel. Fast 3D computations of compressible flow discharge in buildings and complex networks. 2021. <hal-03437352>

**HAL Id: hal-03437352**

**<https://hal.science/hal-03437352v1>**

Preprint submitted on 19 Nov 2021

**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.



HAL Authorization

# Fast 3D computations of compressible flow discharge in buildings and complex networks

Alexandre Chiapolino <sup>(1\*)</sup>, François Fraysse <sup>(1)</sup> and Richard Saurel <sup>(1,2)</sup>

<sup>(1)</sup> RS2N SAS, Saint Zacharie, France

<sup>(2)</sup> Aix Marseille Univ, CNRS, Centrale Marseille, LMA, Marseille, France

## Abstract

Flow computation in complex geometries, such as buildings for example, is challenging at several levels including for instance mesh generation and presence of small elements (doors, windows) that need spatial resolution. When compressible effects are present, such as those resulting from explosions, choking conditions may appear in geometric restrictions. Pressure relaxation to atmospheric condition with existing CFD solvers consequently requires resolved 3D computations, that may be tremendously expensive. In the present work under-resolved computations are addressed, meaning that large computational cells and large time steps are used. Typically, cells' size is of the order of rooms' size. Consequently, special care must be taken at geometric discontinuities. This issue is reminiscent of flow computation with discontinuous area change (Le Floch and Thanh, 2003, Andrianov and Warnecke, 2004, Kroner and Thanh, 2005, Thanh, 2009, Han et al., 2012, to cite a few). However, in the present context, both simplifications and complexities appear. On one hand, full Riemann problem solution is not needed, only flux computation is required resulting in significant simplifications. On the other hand, the flow is 3D, meaning that related complexities must be considered, in the frame of unstructured grids. A fast, efficient, and accurate method is developed for such flows. It consists of two steps. The first step deals with a simple and fast mesh generation where no geometric detail, such as a door for example, is needed. Only “footprints” of the 3D geometry are required. The second step deals with a simple, fast and specific Riemann solver that addresses the previously omitted geometric restrictions directly in the solution states and through the flux distribution as well. The proposed method then appears very convenient when hazardous and pressing situations are involved and require knowledge of the pressure fields. It is validated against resolved computations. Examples are shown with 3D computations of a realistic building.

**Keywords:** Riemann problem, discontinuous area change, choked flow, under-resolved computations, 3D unstructured mesh

<sup>(\*)</sup> Corresponding author [alexandre.chiapolino@rs2n.eu](mailto:alexandre.chiapolino@rs2n.eu) (A. Chiapolino), [francois.fraysse@rs2n.eu](mailto:francois.fraysse@rs2n.eu) (F. Fraysse), [richard.saurel@rs2n.eu](mailto:richard.saurel@rs2n.eu) (R. Saurel)

## I. Introduction

In many situations, determination of the mean (or quasistatic) pressure field is of importance. Relevant examples are pressure fields resulting from explosions in buildings or in complex geometries such as aircrafts, plane wings, industrial plants to cite a few. In these situations, it is obviously possible to build a 3D mesh and compute an appropriate flow model, single-phase or two-phase, depending on the configuration. However, it is time consuming for two main reasons:

- Mesh definition, especially as geometric details are needed, such as for example windows, doors, and various openings. Many thin zones may have first-order effects on the overall flow field.
- Numerical computation of partial differential equations on such domains may be very demanding in computational time and resources. This is typically the case when small openings are present. Small numerical elements are consequently locally needed and affect the global time evolution of the computation.

When both wave propagation and flow discharge effects are strongly coupled it seems to be the only relevant method. However, in most situations, wave propagation rapidly decouples from flow discharge effects. More precisely, computation of blast effects can be accurately achieved with reduced models, based for example on geometrical shock dynamics (Henshaw et al., 1986, Schwendeman, 1993, Whitham, 2011, Ridoux et al., 2018). Simplified methods based on Kingery-Bulmash data are popular engineering alternatives (Kingery, 1966, Coulter et al., 1988, Karlos, 2016). Fast and efficient blast-effect computations have been shown in Frank and Hurley (2005), Lapebie et al. (2016) and Ruscade (2021). These methods couple Kingery-Bulmash data to an algorithm computing shortest distance (Dijkstra 1959) between two points, such as for example the explosive source and a given wall.

The present contribution considers that wave propagation effects are already taken into account through an appropriate method and focuses on flow discharge effects only, having in mind that a fast method is desired, both for the geometry generation step and the flow computation step. Indeed, the present effort attempts to create a simple, accurate and fast method to address hazardous and pressing situations that require knowledge of the mean pressure fields.

To do so, the method uses coarse meshes and few computational cells. This does not seem restrictive for most geometric domains, such as rooms, as only the mean pressure is expected. Moreover, in order to design a simple and fast method generating a geometry and its corresponding mesh, geometric details such as doors for example, are neither drawn nor meshed. Only the 2D “footprints” of the geometry are needed. A conforming 3D mesh with as few elements and limited human time and efforts as possible is then constructed by extruding along the third dimension.

However, using coarse meshes without geometric details requires a specific Riemann solver where geometric restrictions are supposed to be located, such as for example windows and doors. Such Riemann solver is addressed in the present contribution. It is used on elements’ surfaces involving an opening that are only marked during the mesh-generation step.

The proposed Riemann solver addresses the previously omitted geometric restrictions directly in the solution states and through the flux distribution as well. Nevertheless, care is needed as choking conditions may be present at these surfaces and corresponding fluxes must be computed accurately, as pressure distribution is a direct consequence of fluxes’ balance.

Flux computation at these geometric restrictions is reminiscent of the Riemann problem in 1D channels with discontinuous area change (Le Floch and Thanh, 2003, Andrianov and Warnecke, 2004, Kroner and Thanh, 2005, Thanh, 2009, Han et al., 2012).

In the present contribution, the 1D Riemann problem is revisited and analyzed in a simplified situation where right- and left-facing waves are approximated through acoustic relations. Such approximation is sufficiently robust and accurate when dealing with waves of weak amplitude. In the present situation with 3D computations on unstructured meshes, geometric discontinuities (doors and windows) are analogue of 1D discontinuous area change, located at cells' boundaries. It is shown that in the present context, significant simplifications appear in the determination of the Riemann problem solution.

The method with coarse meshes and special treatment of geometric discontinuities is validated against 3D resolved computations. It unsurprisingly appears inaccurate for the handling of wave's dynamics, due to excessive cell size, but appears accurate for the determination of the mean (or quasistatic) pressure field, very important for numerous applications. The pressure relaxation time, a direct consequence of flux computation, is also quite good when dealing with simple geometries. When realistic complex buildings are addressed, the relaxation time appears reasonably accurate, making the present method a simple and very fast numerical tool to address flows in complex buildings.

The paper is organized as follows. Section II deals with mesh generation and specific points simplifying the geometry definition. Section III is devoted to the 1D Riemann problem with discontinuous area change and related flux computation. Section IV deals with validations against resolved computations and 3D computational examples of a realistic building. Conclusions are given in Section V. Finally, an appendix is provided and presents the adaption of the Riemann solver to boundary conditions.

## II. Mesh definition and generation

### II.1 Introduction

The present section addresses a simple and fast method generating a computational geometry and its corresponding mesh. As a fast method, both on the preprocessing stage and on the solver side, is desired, coarse meshes are addressed and the size of a computational cell is typically of the order of the size of a room of a building.

For the sake of simplicity, geometric details, such as doors or windows for example, are neither drawn nor meshed. As will be seen in the following, only the 2D “footprints” of the geometry are needed. A simple linear extraction then provides a conforming 3D mesh.

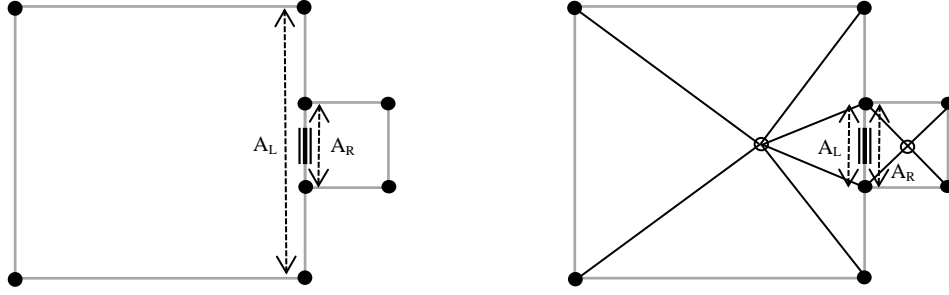
During the present mesh-generation step, surfaces involving specific geometric details are only marked with a flag recognized in a later stage by the fluid flow solver. The geometric restrictions are then considered in a specific Riemann solver, through both its solution states and its flux distribution as well. Section III is devoted to this specific Riemann solver.

It is however necessary to introduce at this stage an important simplification arising from the analysis of the Riemann problem. Such simplification appears when the cross-sections on both sides of a face separating two numerical elements are the same:  $A_L = A_R$ . In this particular situation the geometric restriction, such as a door or a window, becomes transparent in the Riemann problem, at least in the computation of the solution states when an unchoked flow is addressed. Details are provided in Section III.

The present section consequently addresses the construction of a conformal mesh satisfying  $A_L = A_R$  for the marked faces involving a geometric restriction. The other surfaces do not need any particular attention. In such cases, fluxes are provided by a

conventional Riemann solver. The HLLC solver of Toro et al. (1994) is used in the present paper.

The limit case  $\frac{A_R}{A_L} = 1$  is then obtained at all rooms' separations in the present preprocessing stage at the mesh construction level. Figure 1 depicts two possible strategies to construct a mesh from given input nodes.



**Figure 1:** Schematic representation of two strategies to construct a planar mesh for two rooms. On the left side a non-conformal mesh composed of two quadrilaterals is used implying  $A_L \neq A_R$ . On the right side a conformal constrained Delaunay mesh is used yielding  $A_L = A_R$ . In these figures, black filled points represent input nodes. Thick hatched lines represent a door between the two rooms. Black empty circles are additional mesh nodes in the case of a conformal constrained Delaunay mesh.

The first strategy represented in the left side of Figure 1 consists of considering each room as a single discrete element, two quadrilaterals in this simple example. This approach is appealing in the aim of fast computations, due to the minimal element count, but has two strong limitations:

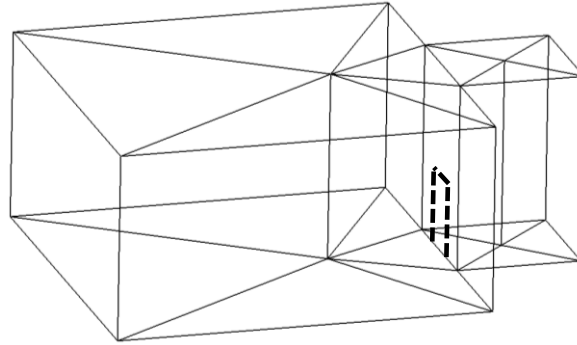
- As the mesh is not conformal, left and right cross-sections are different:  $\frac{A_R}{A_L} \neq 1$ . As a consequence, the above-mentioned simplification of the Riemann solver cannot be used. As will be seen in Section III, this simplification is important and yields a simple, robust, and efficient Riemann solver.  $\frac{A_R}{A_L} \neq 1$  leads consequently to a more complex Riemann solver, affecting potentially both robustness and computational efficiency of the numerical integration;
- Extension to complex 3D configurations is not straightforward, requiring the use of a non-conformal mixed-type element mesh whenever rooms' "footprints" cannot be accurately described by a single quadrilateral element.

The strategy adopted in this work is depicted on the right side of Figure 1. A conformal constrained Delaunay-type mesh is built from given input nodes which ensures  $\frac{A_R}{A_L} = 1$  at all rooms' separations by construction. As detailed in Section III, this condition drastically simplifies the computation of fluxes across doors or windows due to the fact that the geometric restriction becomes transparent in the Riemann problem, when an unchoked flow is addressed.

It may appear at first glance that the simplicity gained on the solver side by considering the limit case  $\frac{A_R}{A_L} = 1$  yields an intricate preprocessing stage with the need of an unstructured meshing strategy. Indeed, the aim is to build a fast numerical framework, both on the preprocessing stage and on the solver side. However, as detailed hereafter, some simplifications can be made on the mesh construction and the flexibility to handle complex 3D configurations obtained by following this strategy is huge.

## II.2 3D Mesh construction

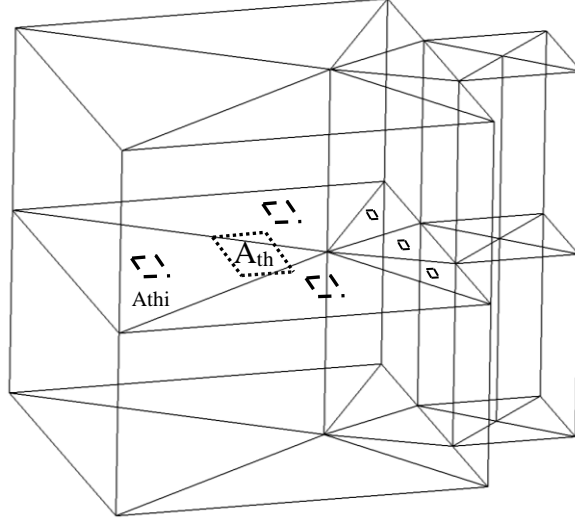
As mentioned in Section II.1 the objective is to construct a conforming mesh with few elements and minimum human effort. The meshing tool used in this work is GMSH (Geuzaine and Remacle, 2009). The strategy adopted here is to construct only the footprint's geometry (nodes, lines and surfaces), generate a 2D mesh and then extrude along the third dimension. In this way, each room is discretized with prismatic elements as can be seen in Figure 2.



**Figure 2:** Two-room prismatic mesh. Fluxes across the separating face (where a door is supposed to be present and represented in thick dashed lines) are computed with the algorithm detailed in Section III and fluxes associated to internal faces are computed with the HLLC Riemann solver.

The separating face is then marked with a flag recognized in a later stage by the fluid flow solver, in the same way boundary conditions are handled in unstructured finite volume codes. The fluid flow software used in this work is the multiphysics DalphiDt® code. DalphiDt is a multi-purpose code handling unstructured meshes composed of arbitrary elements using a cell-centered finite volume method.

One consequence of 2D mesh extrusion is the need of a special treatment to model a geometric restriction between two floors. Indeed, doors and windows belong to a unique quadrilateral face describing a wall. The situation is different with floors for which constrained Delaunay algorithm usually generates several triangles to mesh them. This configuration is depicted in Figure 3. In the rest of the paper, such dimensional reduction will be denoted as the “throat”, in reference to flows occurring in nozzles.



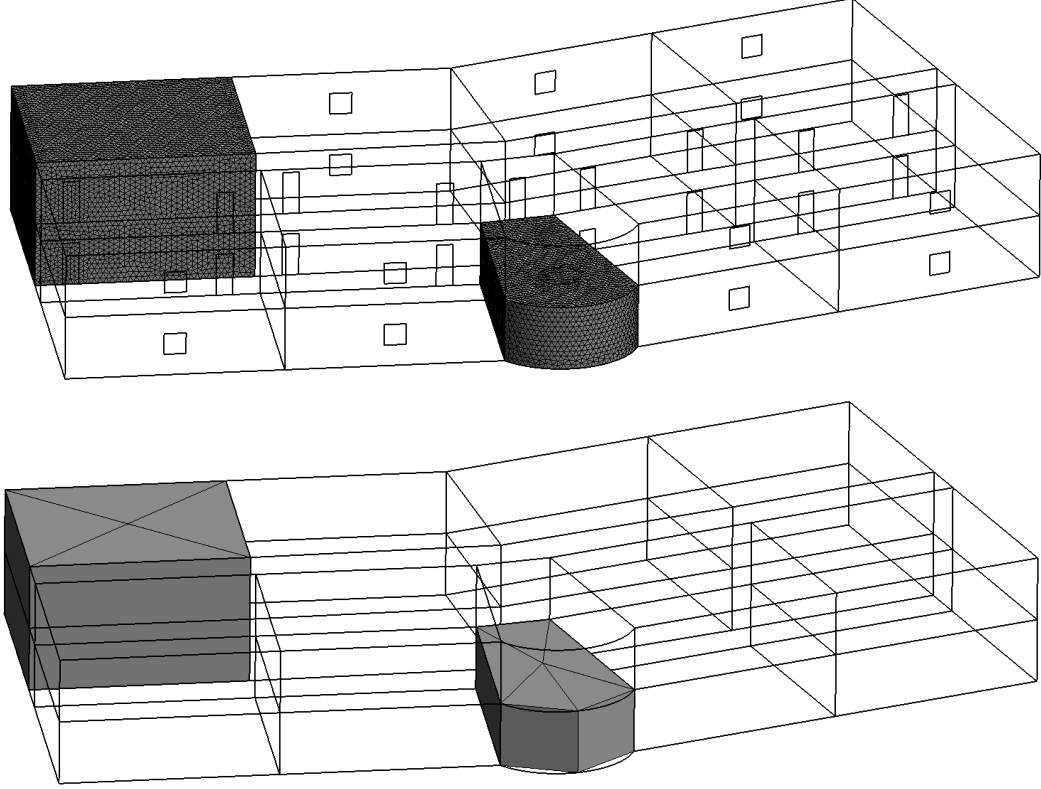
**Figure 3:** Schematic representation of a separation between two floors. The initial throat area  $A_{th}$  (represented with dots), which models for instance a stair, is distributed among all triangular faces pertaining to the floor. Each face then considers a portion  $A_{th_i}$  of the throat area  $A_{th}$ .

Consider a stair between two floors modeled as a throat of area  $A_{th}$ . The ground floor is discretized in  $N$  triangles of area  $S_i, i \in [1, N]$ . The throat area  $A_{th}$  is distributed among all triangular faces pertaining to the ground floor by maintaining a constant aspect ratio:

$$A_{th_i} = A_{th} \frac{S_i}{\sum_{i=1}^N S_i}.$$

In the same spirit, if several doors or windows belong to the same wall, throat areas are merged into a single throat area.

These approximations allow a fast geometry and mesh construction without sacrificing geometric details as can be seen for illustration purpose in Figure 4.



**Figure 4:** Moderately complex building with various rooms, a ground floor and an upper floor. On top, partial view of the conventional 3D tetrahedral mesh where doors, windows and stairs are drawn. On bottom, the simplified prismatic mesh where faces with doors/stairs are treated with the special Riemann solver developed in Section III.

The proposed method ensures  $\frac{A_R}{A_L} = 1$  at all rooms' separations by construction. This property will be used in the following section where a specific Riemann solver is developed to address geometric details (doors, windows, etc.) that have been omitted in the present geometry-definition and mesh-generation preprocessing step.

It is worth mentioning that, as very coarse meshes are used, loss of accuracy appears where curved geometries are considered. This is for example illustrated in Figure 4, where the large numerical cells do not fit the curved portion of the building. One way to remedy to this drawback is to use high-order meshes, see for instance Dobrzynski and Jannoun (2017). Such extension will be examined in the future.

### III. 1D Riemann problem with discontinuous area change and related flux computation

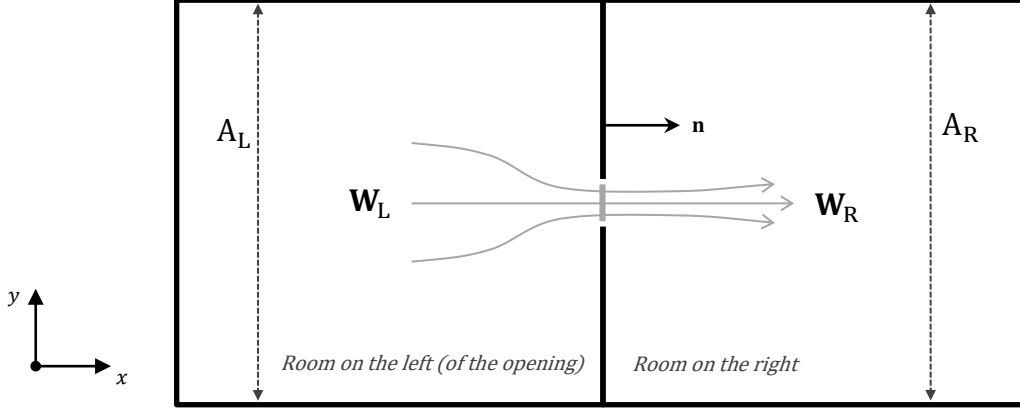
The present section addresses the specific Riemann solver, needed on the marked surfaces of the mesh. Those marked surfaces indicate the presence of a geometric restriction that was not drawn nor meshed during the previous geometry-definition and mesh-generation preprocessing stage. The geometrical effects then need to be considered in the solution states and through the flux distribution of the Riemann solver, as pressure distribution is a direct consequence of the fluxes' balance.

The unmarked faces, where no geometric restriction is present, are treated with the conventional HLLC solver of Toro et al. (1994).



In many situations, determination of the mean (or quasistatic) pressure field is of primary importance. Relevant examples have been mentioned in the Introduction. The pressure field is determined on the basis of the resolution of the Euler equations on coarse 3D unstructured meshes (see Section II).

For the sake of simplicity let us consider a given cell boundary separating two volumes, as shown in Figure 5.

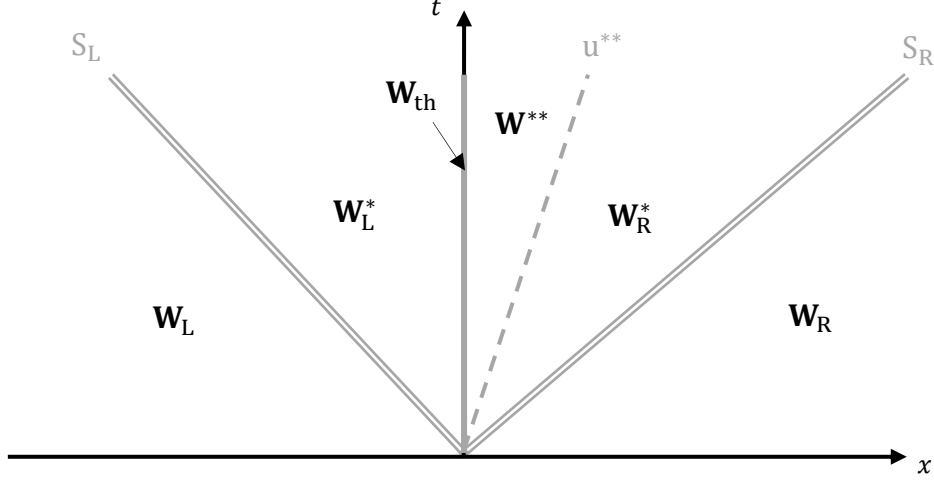


**Figure 5:** Gas flowing from the room on the left of the opening to the room on the right. These two rooms represent computational cells. In the present example, the two rooms have the same cross-section  $A_L = A_R$ . The opening is here only present for illustration purpose. This geometric detail is not meshed. The separating face is only marked with a flag recognized by the fluid flow code (see Section II). A specific Riemann solver is used on such marked faces and addresses the geometric restriction directly in the state solutions and through the flux distribution as well.

As seen in Section II, a conforming mesh with as few elements as possible is used. The opening seen in Figure 5 is neither meshed nor drawn. It only appears for illustration purpose. The rooms on the left and on the right of the geometric discontinuity represent two computational cells. The separating face is only marked with a flag recognized by the fluid flow code, in the same way boundary conditions are handled in unstructured finite volume codes. As seen in the previous section, mesh definition ensures the property  $A_L = A_R$ . As will be seen further, an essential simplification consequently arises.

In Figure 5, the rooms on the left and on the right of the geometric discontinuity represent two computational cells. Consequently, multidimensional effects occurring through the marked face (opening) are not resolved spatially. The main difficulty of the approach dwells at this level, where the dimensional reduction is considered through appropriate quasi-steady relations and assumptions. We recall that a dimensional reduction will be denoted as the “throat”, in reference to flows occurring in nozzles.

The flow evolving between the two rooms is considered through the following Riemann problem, schematized in Figure 6 in the subsonic case.



**Figure 6:** Schematic representation of the wave diagram of 1D Riemann problem between the two rooms separated by a geometric discontinuity (full line). This geometric reduction is assimilated to a throat. Solution of the Riemann problem  $\mathbf{W}_{th}$  is required at this location.  $u^{**}$  represents the contact wave velocity (dashed wave) while  $S_L$  and  $S_R$  represent the left- and right-facing wave speeds (double-lined waves). The present waves are a consequence of the eigenvalues of the hyperbolic 1D Euler equations for ducts of variable cross-sections.

The analysis is carried out at this level with the 1D Euler equations for ducts of variable cross-sections:

$$\left\{ \begin{array}{l} \frac{\partial A}{\partial t} = 0, \\ \frac{\partial(\rho A)}{\partial t} + \frac{\partial(\rho u A)}{\partial x} = 0, \\ \frac{\partial(\rho u A)}{\partial t} + \frac{\partial((\rho u^2 + p) A)}{\partial x} = p \frac{\partial A}{\partial x}, \\ \frac{\partial(\rho E A)}{\partial t} + \frac{\partial((\rho E + p) u A)}{\partial x} = 0. \end{array} \right. \quad (\text{III.1})$$

The notations are conventional.  $x$  is the position and  $t$  is the time.  $A$  denotes the cross-section.  $\rho$ ,  $u$  and  $E$  represent respectively the density, the velocity and the total energy defined as:  $E = e + \frac{1}{2}u^2$  where  $e$  is the internal energy. The pressure  $p$  is computed with the ideal-gas equation of state,

$$p = (\gamma - 1)\rho e,$$

where  $\gamma$  represents the isentropic exponent, or ratio of the specific heats.

In Figure 6,  $\mathbf{W}$  represents the vector of primitive variables  $\mathbf{W} = (\rho, u, p, A)^T$ . The cross-section  $A$  varies only between states  $\mathbf{W}_L^*$  and  $\mathbf{W}^{**}$  of this figure. The flow model (III.1) admits the following additional entropy (denoted  $s$ ) equation:

$$\frac{\partial(\rho s A)}{\partial t} + \frac{\partial(\rho s u A)}{\partial x} = 0. \quad (\text{III.2})$$

This equation is particularly important in the present context. The analysis begins with the determination of the flow direction. Then, unchoked and choked conditions at the throat are analyzed separately and two specific procedures are developed. A method allowing to determine the appropriate flow regime, in accordance with the flow conditions, is afterwards presented. The section ends by summarizing the global method determining the solution state of the Riemann problem, and by introducing the specific flux computation.

### III.1 Flow direction determination

The Riemann problem for the Euler equations, without cross-section variation, is solved in the fluid cross-section, at the throat, between two fluid states. This is the same type of computation done locally with a multidimensional code. More precisely the 1D Euler equations read,

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = 0, \quad (\text{III.1.1})$$

with  $\mathbf{U} = (\rho, \rho u, \rho E)^T$  and  $\mathbf{F} = (\rho u, \rho u^2 + p, (\rho E + p)u)^T$ . The Riemann problem is solved under the HLL approximation (Harten et al., 1983) with:

$$\mathbf{U}^* = \frac{\mathbf{F}_R - \mathbf{F}_L + S_L \mathbf{U}_L - S_R \mathbf{U}_R}{S_L - S_R}, \quad (\text{III.1.2})$$

where states  $\mathbf{L}$  and  $\mathbf{R}$  are the same as those of Figure 6. The two extreme wave speeds  $S_L$  and  $S_R$  are provided with the help of Davis' (1988) estimates.

The second component of vector  $\mathbf{U}^*$ , corresponding to the momentum equation,  $U_{\text{mom}} = \rho u$ , provides the flow direction. This flow direction is important for the determination of the critical state, associated to sonic conditions at the throat as will be seen later. First let us consider unchoked flow conditions at the throat.

### III.2 Unchoked flow

In the present section, the flow at the throat is unchoked. In such situation, the flow does not change regime. That is to say that if the flow upstream is subsonic, it remains subsonic at the throat. The same can be said for a supersonic flow. For the sake of simplicity, let us begin the analysis in the subsonic case.

#### Subsonic flow with $u^{**} > 0$

This situation corresponds precisely to the one depicted in Figure 6. It is a specific flow configuration allowing gradual consideration of the various effects. To simplify the calculations two main assumptions are done:

- The left- and right-facing waves are considered under acoustic approximations. It means that Rankine-Hugoniot relations as well as Riemann invariants are replaced by characteristic equations with constant acoustic impedance.
- The isentropes, or more precisely Laplace's law, is approximated by the sound speed definition. It corresponds to a linear approximation of the isentropes.

These two approximations are conventional and given for example in Toro (1997) in the frame of the approximate acoustic Riemann solver. They are valid for waves of weak amplitude, which is appropriate to the present context of subsonic evolutions. The set of relations to consider is summarized hereafter.

For a left-facing wave, the approximations are:

$$p_L + Z_L u_L = p_L^* + Z_L u_L^*, \quad (\text{III.2.1})$$

$$\rho_L^* = \rho_L + \frac{p_L^* - p_L}{c_L^2}. \quad (\text{III.2.2})$$

For a right-facing wave, the approximations are:

$$p_R - Z_R u_R = p_R^* - Z_R u_R^*, \quad (\text{III.2.3})$$

$$\rho_R^* = \rho_R + \frac{p_R^* - p_R}{c_R^2}. \quad (\text{III.2.4})$$

In these relations  $Z = \rho c$  represents the acoustic impedance, where  $c = \left( \gamma \frac{p}{\rho} \right)^{1/2}$  represents the sound speed, according to the ideal-gas equation of state.

The contact wave is governed by interface conditions:

$$u_R^* = u^{**}, \quad (\text{III.2.5})$$

$$p_R^* = p^{**}. \quad (\text{III.2.6})$$

The flow is assumed subsonic everywhere and in particular between states  $\mathbf{W}_L^*$  and  $\mathbf{W}^{**}$ . Moreover, the flow is assumed stationary between these two states. Under this latter condition, the integration of the equations of System (III.1) and Equation (III.2) between states  $\mathbf{W}_L^*$  and  $\mathbf{W}^{**}$  results in the conservation of the mass flow rate, the conservation of the specific total enthalpy and the conservation of the specific entropy. The first two relations read:

$$\rho_L^* u_L^* A_L = \rho^{**} u^{**} A_R, \quad (\text{III.2.7})$$

$$\frac{\gamma p^{**}}{(\gamma-1)\rho^{**}} + \frac{1}{2} u^{**2} = \frac{\gamma p_L^*}{(\gamma-1)\rho_L^*} + \frac{1}{2} u_L^{*2}, \quad (\text{III.2.8})$$

where the ideal-gas equation of state has been introduced. As mentioned earlier, the conservation of the specific entropy is here approximated by the sound speed definition, corresponding to a linear approximation of the isentropes:

$$\rho^{**} = \rho_L^* + \frac{p^{**} - p_L^*}{c_L^{*2}}. \quad (\text{III.2.9})$$

Each state,  $\mathbf{W}_L^*$ ,  $\mathbf{W}^{**}$  and  $\mathbf{W}_R^*$ , contains 3 unknowns, corresponding to a total number of 9 unknowns. The algebraic system (III.2.1 – III.2.9) involves 9 algebraic equations. Consequently, the system is closed. Its resolution is done as follows:

- Arbitrary guess of  $p^{**}$  is set, implying  $p_R^* = p^{**}$ .
- With the help of (III.2.3) and (III.2.4) the following variables are deduced:

$$u^{**} = u_R + \frac{p^{**} - p_R}{Z_R} \quad (\text{III.2.10})$$

$$\rho_R^* = \rho_R + \frac{p^{**} - p_R}{c_R^2} \quad (\text{III.2.11})$$

At this level, full state  $\mathbf{W}_R^*$  is determined.

- In the  $\mathbf{W}^{**}$  state,  $p^{**}$  has been set and (III.2.10) provides  $u^{**}$ . It remains to determine  $\rho^{**}$ . Combining (III.2.9) and (III.2.2) yields:

$$\rho^{**} = \rho_L + \frac{p_L^* - p_L}{c_L^2} + \frac{p^{**} - p_L^*}{c_L^{*2}}.$$

Extra simplification can be done, assuming  $c_L^2 = c_L^{*2}$ . In the present subsonic conditions, it appears reasonable. The previous relation thus transforms to,

$$\rho^{**} = \rho_L + \frac{p^{**} - p_L}{c_L^2} \quad (\text{III.2.12})$$

At this level, full state  $\mathbf{W}^{**}$  is determined.

- System (III.2.7 – III.2.8) is considered for the determination of  $\mathbf{W}_L^*$ . Combining these relations, the following one is obtained,

$$\left( \gamma c_L^2 - \frac{\gamma p^{**}}{\rho^{**}} - \frac{1}{2}(\gamma - 1)u^{*2} \right) \rho_L^{*2} + \gamma (p^{**} - c_L^2 \rho^{**}) \rho_L^* + \frac{(\gamma - 1)}{2} \left( \frac{A_R}{A_L} \right)^2 \rho^{*2} u^{*2} = 0. \quad (\text{III.2.13})$$

The positive root of this quadratic equation is retained, giving  $\rho_L^*$ . With the help of (III.2.2) the pressure  $p_L^*$  is determined:

$$p_L^* = p^{**} - c_L^2 (\rho^{**} - \rho_L^*).$$

Then, using (III.2.7) the velocity is determined,

$$u_L^* = \frac{\rho^{**} u^{**} A_R}{\rho_L^* A_L}.$$

State  $\mathbf{W}_L^*$  is now fully determined.

However, the  $\mathbf{W}_L^*$  solution state is not necessarily compatible with Relation (III.2.1). If the function,

$$f(\mathbf{p}^{**}) = p_L + Z_L u_L - (p_L^* + Z_L u_L^*),$$

has not reached a certain tolerance, the initial pressure  $p^{**}$  must be changed until this condition is reached. Such task can be done with Newton's method. It is certainly possible to optimize and generalize this algorithm both for  $u^{**} > 0$  and  $u^{**} < 0$ . However, it does not seem important in the present context, because of the limit case situation that follows.

**Limit case**  $\frac{A_R}{A_L} = 1$

This limit case corresponds for instance to the situation depicted in Figure 5 where the two rooms have the same cross-section. It is particularly important as, during mesh generation (see Section II), these two surfaces merge when the cell boundary includes an opening (door or window). In other words, cells' faces merge on surface boundaries containing an opening during 3D meshing. Relation (III.2.13) becomes:

$$\left( \gamma c_L^2 - \frac{\gamma p^{**}}{\rho^{**}} - \frac{1}{2}(\gamma - 1)u^{**2} \right) \rho_L^{*2} + \gamma (p^{**} - c_L^2 \rho^{**}) \rho_L^* + \frac{(\gamma - 1)}{2} \rho^{**2} u^{**2} = 0,$$

i.e.,

$$\gamma \rho_L^* \left( c_L^2 - \frac{p^{**}}{\rho^{**}} \right) (\rho_L^* - \rho^{**}) + \frac{(\gamma - 1)}{2} u^{**2} (\rho^{**2} - \rho_L^{*2}) = 0.$$

After simplifications it implies,

$$\rho_L^* = \rho^{**}.$$

Then, considering mass conservation (III.2.7)  $m = \rho^{**} u^{**} A_R$ , it results:

$$u_L^* = u^{**}.$$

It then follows that  $p_L^* = p^{**}$ , from enthalpy invariance (III.2.8).

This result is not surprising but is very important. As the flow is isentropic and preserves mass and energy fluxes, the varying cross-section has no effect, at least in the computation of the states of the Riemann problem. This is true in the present case of a subsonic flow everywhere, even at the throat.

A relevant simplification appears. As the geometric discontinuity is transparent ( $\mathbf{W}_L^* = \mathbf{W}^{**}$ ) there is no need to consider 4 waves and 3 states in the Riemann problem. In this particular case it reduces to 3 waves and 2 states, as done usually with the Euler equations, without cross-section variation. In this frame, the acoustic solver is no longer used and replaced by the more general and more robust HLLC solver of Toro et al. (1994).

This simplification is valid whatever the sign of  $u^{**}$  is, provided that the flow is not choked at the throat. It is also valid for supersonic flows, provided that the flow is unchoked at the throat.

Consequently, as seen in Section II, only situations involving  $\frac{A_R}{A_L} = 1$  on a marked cell's face presenting a geometric restriction (door, window, etc.) are considered in the frame of the present method.

The HLLC solver is then used and provides the  $\mathbf{W}_L^*$  state of Figure 6. From the  $\mathbf{W}_L^*$  state, the state at the throat  $\mathbf{W}_{th}$  is then computed. Its determination is based on the same algebraic system as (III.2.7 – III.2.9), except that the conservation of the specific entropy is now expressed through Laplace's law. Knowledge of the state at the throat  $\mathbf{W}_{th}$  is important for two reasons:

- To check the validity of the solution. Indeed, the solution from HLLC solver is valid only when the opening between the two rooms is transparent. This is correct only if the state at the throat is unchoked. It is then necessary to check whether this state is choked or not. This point is addressed in Section III.4.
- To determine the effective fluxes that cross the cell boundary through the opening. These fluxes are significantly different from those associated to state  $\mathbf{W}_L^*$ .

Determination of the solution state at the throat  $\mathbf{W}_{th}$  is addressed in the following, as in the present paragraph, the flow is assumed subsonic everywhere, even at the throat.

The method is presented according to the flow situation depicted in Figure 6. The solution speed is positive ( $U_{mom}^* > 0$ ) and the HLLC solver is assumed to provide the solution state  $\mathbf{W}_L^*$ . Naturally the method presented hereafter treats the opposite situation ( $U_{mom}^* < 0$ ) similarly. In that case the HLLC solver provides the solution state  $\mathbf{W}_R^*$ .

Between a given state  $\mathbf{W}_L^* = (\rho_L^*, u_L^*, p_L^*, A_L^* = A_L)^T$  and the state at the throat  $\mathbf{W}_{th}$ , the corresponding equations read:

$$\rho_L^* u_L^* A_L^* = \rho_{th} u_{th} A_{th}, \quad (III.2.14)$$

$$\frac{p_L^*}{\rho_L^{*\gamma}} = \frac{p_{th}}{\rho_{th}^\gamma}, \quad (III.2.15)$$

$$\frac{\gamma p_L^*}{(\gamma-1)\rho_L^*} + \frac{1}{2} u_L^{*2} = \frac{\gamma p_{th}}{(\gamma-1)\rho_{th}} + \frac{1}{2} u_{th}^2. \quad (III.2.16)$$

With the help of the ideal-gas sound speed definition, the specific total enthalpy relation (III.2.16) yields, after some algebraic manipulations:

$$1 + \frac{\gamma-1}{2} M_L^{*2} = \frac{c_{th}^2}{c_L^{*2}} \left( 1 + \frac{\gamma-1}{2} M_{th}^2 \right), \quad (III.2.17)$$

where  $M = u/c$  is the Mach number. Similar manipulations on Laplace's law (III.2.15) leads to,

$$\frac{\rho_{th}}{\rho_L^*} = \left( \frac{c_{th}}{c_L^*} \right)^{\frac{2}{\gamma-1}}. \quad (III.2.18)$$

The mass conservation (III.2.14) is now written under the following form,

$$\frac{\rho_{th}}{\rho_L^*} = \frac{A_L^* M_L^* c_L^*}{A_{th} M_{th} c_{th}}. \quad (III.2.19)$$

The equality of Relations (III.2.18) and (III.2.19) yields,

$$\left( \frac{c_{th}}{c_L^*} \right)^2 = \left( \frac{A_L^* M_L^*}{A_{th} M_{th}} \right)^{\frac{2(\gamma-1)}{\gamma+1}}. \quad (III.2.20)$$

This last relation is now inserted into Relation (III.2.17), leading to:

$$\frac{A_L^*}{A_{th}} = \frac{M_{th}}{M_L^*} \left( \frac{1 + \frac{(\gamma-1)}{2} M_L^{*2}}{1 + \frac{(\gamma-1)}{2} M_{th}^2} \right)^{\frac{\gamma+1}{2(\gamma-1)}}. \quad (III.2.21)$$

The Mach number at the throat  $M_{th}$  is determined from this last relation. An iterative method is required though. Then, the sound speed at the throat  $c_{th}$  is deduced from Relation (III.2.20). The corresponding velocity  $u_{th}$  is readily obtained from the Mach number definition  $M_{th} = u_{th} / c_{th}$ . The density  $\rho_{th}$  is afterwards determined from Relation (III.2.18). Finally, the pressure  $p_{th}$  is computed from Laplace's law (III.2.15). The whole state at the throat  $\mathbf{W}_{th}$  is then fully determined for an unchoked flow at the throat.

Note that when  $A_L = A_{th}$ , Relation (III.2.21) involves  $M_{th} = M_L^*$ . In such case, the whole HLLC solution is recovered  $\mathbf{W}_{th} = \mathbf{W}_L^*$ , through Relations (III.2.20), (III.2.19) and (III.2.15), and the stationary wave occurring at the throat (Figure 6) disappears.

The following section is devoted to a choked flow, a specific situation where sonic conditions are met at the throat. Such sonic situation only applies when an opening is present in the Riemann problem. The following section does not apply in the limit situation where no dimensional reduction occurs, i.e.,  $A_L = A_{th}$ . In such case, the HLLC solver shall be used directly.

### III.3 Sonic flow

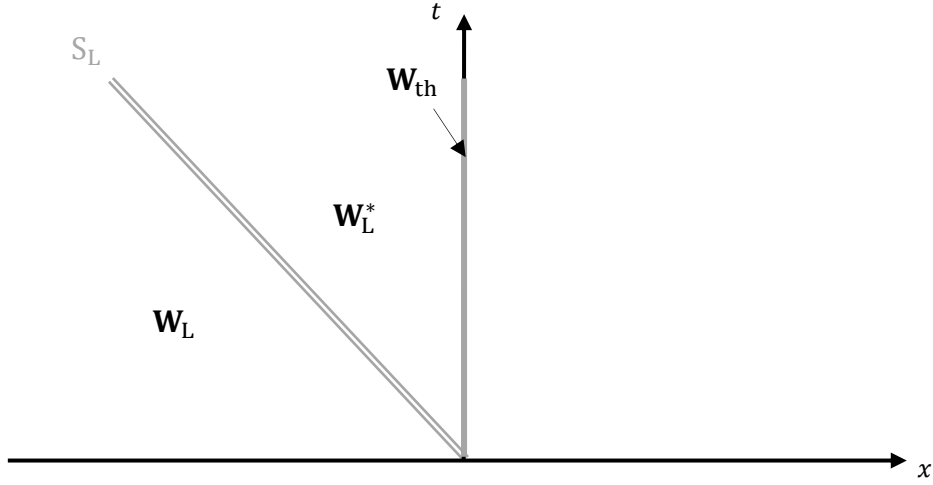
The present section deals with sonic conditions at the throat. The flow is consequently choked and a specific resolution is needed. Indeed, as the flow Mach number reaches unity at the throat, pressure disturbances can no longer be communicated upstream. Consequently, the upstream side is isolated from the downstream side at the throat.

A specific Riemann problem must then be solved. This specific Riemann problem only accounts for one side of the throat. More precisely, full Riemann problem resolution is not addressed, only flux computation at cell's boundary is needed, including an opening (door or window) as shown in Figure 5, where the flow is now choked.



A part of the surface acts as a reflective wall (Figure 5). Consequently, a reflected wave affects the solution. To illustrate the situation, let us imagine a subsonic flow in a state  $\mathbf{W}$  and a wall with a small hole. In the hole cross-section, the flow is sonic. Critical conditions are then reached. But as the main part of the cross-section is closed by the wall, a reflected wave propagates into state  $\mathbf{W}^*$ . The amplitude of the reflected wave is such that the modified state ( $\mathbf{W}^*$ ) is associated to a new critical state such that the area at the throat becomes strictly equal to the new critical area:  $A_{cr}^{new} = A_{th}$ . We will come back to the critical state determination later, when determining the flow regime, i.e. choked or unchoked.

When sonic conditions are met, the upstream side is isolated from the downstream side at the throat. The situation depicted in Figure 6 then transforms to the situation depicted in Figure 7, representing the present sonic half Riemann problem.



**Figure 7:** Schematic representation of the wave diagram of 1D Riemann problem between the two rooms separated by a geometric discontinuity (full line). This geometric reduction is considered as a throat. Solution of the Riemann problem  $\mathbf{W}_{th}$  is required at this location. In the present situation, the flow is sonic at the throat. Choked conditions then appear and isolate the upstream side from the downstream side at the throat. As the surface on which the Riemann problem is solved is a wall containing an opening, a reflected wave appears and affects the solution.

As previously, let us present the method according to the flow situation depicted in Figure 7. The solution speed is then positive, according to the HLL solution as discussed in Section III.1 ( $U_{mom}^* > 0$ ), and the two solution states to be determined in the present specific Riemann problem are  $\mathbf{W}_L^*$  and  $\mathbf{W}_{th}$ . Naturally the method presented hereafter treats the opposite situation ( $U_{mom}^* < 0$ ) similarly. In that case the two solution states to be determined in the present specific Riemann problem are  $\mathbf{W}_R^*$  and  $\mathbf{W}_{th}$ .

However, we will now include the subscript “sonic” to specify that the following method applies only in the specific situation where sonic conditions are met at the throat. We will then denote the corresponding solution states by  $\mathbf{W}_{L,sonic}^*$  and  $\mathbf{W}_{th,sonic}$ . It must be stressed that  $\mathbf{W}_{L,sonic}^*$  represents the solution state that leads, through an isentropic process, to sonic conditions at the throat.  $\mathbf{W}_{L,sonic}^*$  does not involve Mach number equal to unity. The sonic condition  $M_{th,sonic} = 1$  only applies at the throat.

For the sake of simplicity, the reflected wave from the permeable wall is considered through acoustic approximation:

$$p_{L,\text{sonic}}^* = p_L + Z_L (u_L - u_{L,\text{sonic}}^*), \quad (\text{III.3.1})$$

$$\rho_{L,\text{sonic}}^* = \rho_L + \frac{p_{L,\text{sonic}}^* - p_L}{c_L^2}. \quad (\text{III.3.2})$$

The characteristic relation (III.3.1) assumes constant acoustic impedance  $Z_L = \rho_L c_L$  across the reflected wave. Such assumption is valid for waves of weak amplitude and is considered for the present analysis. The sign “+” applies for a left-facing reflected wave. When the flow is reversed ( $U_{\text{mom}}^* < 0$ ), sign “-” applies,

$$p_{R,\text{sonic}}^* = p_R - Z_R (u_R - u_{R,\text{sonic}}^*).$$

Relation (III.3.2) is based on the sound speed definition and is a linearized version of Laplace’s law. Then, between state  $\mathbf{W}_{L,\text{sonic}}^*$  and the sonic throat  $\mathbf{W}_{\text{th,sonic}}$ , the same system as previously (III.2.14 – III.2.16) holds. However sonic conditions apply in addition. The corresponding system consequently reads,

$$\rho_{L,\text{sonic}}^* u_{L,\text{sonic}}^* A_L^* = \rho_{\text{th,sonic}} u_{\text{th,sonic}} A_{\text{th}}, \quad (\text{III.3.3})$$

$$\frac{p_{L,\text{sonic}}^*}{\rho_{L,\text{sonic}}^{*\gamma}} = \frac{p_{\text{th,sonic}}}{\rho_{\text{th,sonic}}^\gamma}, \quad (\text{III.3.4})$$

$$\frac{\gamma p_{L,\text{sonic}}^*}{(\gamma-1)\rho_{L,\text{sonic}}^*} + \frac{1}{2} u_{L,\text{sonic}}^{*2} = \frac{\gamma p_{\text{th,sonic}}}{(\gamma-1)\rho_{\text{th,sonic}}} + \frac{1}{2} u_{\text{th,sonic}}^2, \quad (\text{III.3.5})$$

$$u_{\text{th,sonic}}^2 = c_{\text{th,sonic}}^2. \quad (\text{III.3.6})$$

System (III.3.3 – III.3.6) involves the unknowns  $p_{L,\text{sonic}}^*$ ,  $u_{L,\text{sonic}}^*$ ,  $\rho_{L,\text{sonic}}^*$ ,  $\rho_{\text{th,sonic}}$ ,  $u_{\text{th,sonic}}$  and  $p_{\text{th,sonic}}$ . The geometric areas  $A_L^* = A_L$  and  $A_{\text{th}}$  are perfectly known at this level. System (III.3.3 – III.3.6) is consequently closed with the help of Relations (III.3.1-III.3.2).

As seen in the previous section, the combination of the mass equation (III.3.3), Laplace’s law (III.3.4) and enthalpy equation (III.3.5) yields Relation (III.2.21), linking the Mach numbers to the geometric areas. The present section deals with sonic conditions at the throat where  $M_{\text{th,sonic}} = 1$  appears. In such conditions, Relation (III.2.21) reduces to,

$$\frac{A_L}{A_{\text{th}}} = \frac{1}{M_{L,\text{sonic}}^*} \left( \frac{2}{\gamma+1} \left( 1 + \frac{\gamma-1}{2} M_{L,\text{sonic}}^{*2} \right) \right)^{\frac{\gamma+1}{2(\gamma-1)}}. \quad (\text{III.3.7})$$

Recall that  $M_{L,\text{sonic}}^*$  represents the Mach number in the  $\mathbf{W}_{L,\text{sonic}}^*$  state when this state is meant to bring sonic conditions at the throat.  $M_{L,\text{sonic}}^*$  is then different from 1. Note that the previous relation implies that the throat cross-section is the critical area,  $A_{\text{cr}} = A_{\text{th}}$ . We will come back to the critical condition in the next section.

Relation (III.3.7) is solved to determine  $M_{L,\text{sonic}}^*$ . An iterative method is required though. Then using the Mach number definition,

$$u_{L,\text{sonic}}^* = M_{L,\text{sonic}}^* c_{L,\text{sonic}}^* (p_{L,\text{sonic}}^*, \rho_{L,\text{sonic}}^*), \quad (\text{III.3.8})$$

and upon insertion of (III.3.1 – III.3.2), and the ideal-gas equation of state  $c = \left( \gamma \frac{p}{\rho} \right)^{1/2}$ , it becomes,

$$u_{L,\text{sonic}}^* = M_{L,\text{sonic}}^* \sqrt{\gamma \frac{p_L + Z_L(u_L - u_{L,\text{sonic}}^*)}{\rho_L \left( 1 + \frac{u_L - u_{L,\text{sonic}}^*}{c_L} \right)}}. \quad (\text{III.3.9})$$

Relation (III.3.9) yields a nonlinear function, requiring another iterative process to compute  $u_{L,\text{sonic}}^*$ . Note that when the flow is reversed ( $U_{\text{mom}}^* < 0$ ), sign “-” applies,

$$u_{R,\text{sonic}}^* = M_{R,\text{sonic}}^* \sqrt{\gamma \frac{p_R - Z_R(u_R - u_{R,\text{sonic}}^*)}{\rho_R \left( 1 - \frac{u_R - u_{R,\text{sonic}}^*}{c_R} \right)}}.$$

Once the velocity  $u_{L,\text{sonic}}^*$  is determined, pressure  $p_{L,\text{sonic}}^*$  and density  $\rho_{L,\text{sonic}}^*$  are computed by Relations (III.3.1 – III.3.2). The flow variables at the throat  $\mathbf{W}_{\text{th,sonic}}$  are then deduced from Relations (III.3.3 – III.3.6) leading to,

$$c_{\text{th,sonic}}^2 = u_{\text{th,sonic}}^2 = \frac{2}{\gamma + 1} \left( \frac{\gamma p_{L,\text{sonic}}^*}{\rho_{L,\text{sonic}}^*} + \frac{1}{2} (\gamma - 1) u_{L,\text{sonic}}^{*2} \right), \quad (\text{III.3.10})$$

$$\rho_{\text{th,sonic}} = \rho_{L,\text{sonic}}^* \left[ \frac{2}{\gamma + 1} \left( 1 + \frac{\gamma - 1}{2} M_{L,\text{sonic}}^{*2} \right) \right]^{\frac{1}{\gamma - 1}}, \quad (\text{III.3.11})$$

$$p_{\text{th,sonic}} = p_{L,\text{sonic}}^* \frac{2}{\gamma + 1} \left( \frac{p_{L,\text{sonic}}^*}{\rho_{L,\text{sonic}}^*} + \frac{1}{2} \frac{\gamma - 1}{\gamma} u_{L,\text{sonic}}^{*2} \right). \quad (\text{III.3.12})$$

The fluxes for the mass, momentum and energy equations are computed with this set of variables, as detailed later. First let us come back to the critical state allowing to select the regime, subsonic or sonic, appropriate to the flow conditions.

### III.4 Flow regime

Two specific procedures have been developed previously. The first deals with an unchoked flow at the throat and the second deals with choked conditions. The present section addresses selection of the appropriate flow regime according to the flow conditions.

The method begins by assuming subsonic (unchoked) flow everywhere, even at the throat. This assumption is then assessed by comparing the subsonic solution to critical conditions.

The subsonic Riemann problem depicted in Figure 6 is then considered and used one more time to present the proposed method. The flow direction is assumed “positive”, i.e.,  $u^{**} > 0$  and the  $\mathbf{W}_L^*$  solution state is used. Naturally, the method similarly applies for the reverse situation, i.e.  $u^{**} < 0$ .

As seen in Section III.2, assuming subsonic flow at the throat yields simple, robust and direct computation of the  $\mathbf{W}_L^*$  solution state with the help of the HLLC solver. Recall that we only deal with situations involving  $A_L = A_R$ . Computation of the solution state at the throat  $\mathbf{W}_{th}$  or the solution states in sonic conditions  $\mathbf{W}_{L,sonic}^*, \mathbf{W}_{th,sonic}$  is not direct and requires iterative resolutions (Section III.2 and III.3).

The subsonic  $\mathbf{W}_L^*$  solution state, resulting from the HLLC solver, will then be used to determine the flow regime at the throat, i.e., subsonic or sonic, and consequently use the Riemann solver appropriate to the situation.

#### III.4.1 Critical state

The analysis of the 1D equations of compressible fluid mechanics, in stationary and isentropic situations, reveals two conditions for the throat to be choked:

$$\begin{cases} A_{th} \leq A_{cr}, \\ R_p \leq R_{pcr}, \end{cases} \rightarrow \text{sonic.} \quad (\text{III.4.1})$$

The first condition involves the geometric throat cross-section  $A_{th}$  that becomes equal or less than the critical area  $A_{cr}$  when sonic conditions are met at the throat. The second condition involves the ratio between the static and stagnation pressures at the throat  $R_p$  that must also be equal or less than the critical ratio  $R_{pcr}$ .

In the present context, the two properties: stationary and isentropic, appear across the stationary wave, that is to say between states  $\mathbf{W}_L^*$  and  $\mathbf{W}^{**}$  of Figure 6. The proposed method begins by assuming subsonic flow. The assessment of the subsonic assumption must then be carried out between the subsonic solution state  $\mathbf{W}_L^*$  and the  $\mathbf{W}_{th}$  solution state at the throat. Relevance of the subsonic assumption is examined through inequalities:

$$\begin{cases} A_{th} > A_{cr}, \\ R_p > R_{pcr}, \end{cases} \rightarrow \text{subsonic.} \quad (\text{III.4.2})$$

Assuming subsonic flow yields direct computation of the  $\mathbf{W}_L^*$  solution state via the HLLC solver. If such solution satisfies both inequalities of (III.4.2) then the subsonic assumption is relevant and the  $\mathbf{W}_{th}$  solution state is computed with the unchoked Riemann solver presented in Section III.2. However, if inequalities of (III.4.2) are not fulfilled, the subsonic solution  $\mathbf{W}_L^*$  must be left out as choked conditions appear at the throat. The specific Riemann solver presented in Section III.3 is then used and provides the actual  $\mathbf{W}_{L,sonic}^*$  solution state as well as the  $\mathbf{W}_{th,sonic}$  solution state involving sonic conditions at the throat.

The method then requires knowledge of  $A_{th}$ ,  $A_{cr}$  and  $R_p$ ,  $R_{pcr}$  from the subsonic solution state  $\mathbf{W}_L^*$  to determine the flow regime. Let us start by analyzing the critical area.

### Critical area

The critical area  $A_{cr}$  represents the fictitious minimum throat cross-section that would be necessary to isentropically accelerate or decelerate the flow to a Mach number of 1. Its expression results from the conservation of mass, specific entropy, and specific total enthalpy between states  $\mathbf{W}_L^*$  and  $\mathbf{W}_{th}$  in addition to the sonic condition  $u_{th} = c_{th}$  at the throat. The combination of those last points yields Eq. (III.3.7), developed during the analysis of the choked situation in Section III.3, and reformulated hereafter as:

$$A_{cr} = A_L M_L^* \left( \frac{1 + \frac{(\gamma-1)}{2} M_L^{2*}}{1 + \frac{(\gamma-1)}{2}} \right)^{-\frac{\gamma+1}{2(\gamma-1)}}. \quad (III.4.3)$$

The critical area  $A_{cr}$  is then known from Eq. (III.4.3) and the subsonic solution state  $\mathbf{W}_L^*$  obtained from the HLLC solver. It provides the fictitious throat area  $A_{cr}$ , related to the current  $\mathbf{W}_L^*$  state, for choked flow conditions to appear. As long as  $A_{th} > A_{cr}$ , the flow does not change regime. However, the previous inequality is not the only condition to be satisfied for the subsonic assumption to be relevant. The ratio  $R_p$  between the static and stagnation pressures must also be compared to the critical one  $R_{pcr}$ .

### Critical pressure ratio

The critical pressure ratio  $R_{pcr}$  represents the fictitious minimum ratio between the static and stagnation pressures at the throat required for the flow to become choked. It reads:

$$R_{pcr} = \frac{p_{th,sonic}}{p_{0,sonic}}, \quad (III.4.4)$$

where the subscript “sonic” has been one more time added to specify that  $p_{th,sonic}$  and  $p_{0,sonic}$  represent respectively the static pressure at the throat and the stagnation pressure when sonic conditions are met at the throat. In the following the specification “sonic” will be used every time sonic conditions are involved.

The stagnation pressure describes the fictitious pressure of a fluid adiabatically brought to rest. Its expression results from the invariance of specific entropy and specific total enthalpy between states  $\mathbf{W}_{L,sonic}^*$  and  $\mathbf{W}_{th,sonic}$ . The analysis shows that the stagnation pressure does not vary throughout an isentropic flow. With the help of the ideal-gas equation of state, the stagnation pressure reads:

$$p_{0,\text{sonic}} = p_{\text{th,sonic}} \left( 1 + \frac{\gamma-1}{2} \underbrace{M_{\text{th,sonic}}^2}_1 \right)^{\frac{\gamma}{\gamma-1}} = p_{\text{L,sonic}}^* \left( 1 + \frac{\gamma-1}{2} M_{\text{L,sonic}}^{*2} \right)^{\frac{\gamma}{\gamma-1}}. \quad (\text{III.4.5})$$

Again, it is important to stress that  $p_{\text{L,sonic}}^*$  and  $M_{\text{L,sonic}}^*$  represent the pressure and Mach number in the  $\mathbf{W}_{\text{L,sonic}}^*$  state when this state is meant to bring sonic conditions at the throat.  $M_{\text{L,sonic}}^*$  is then different from 1. As sonic conditions are considered  $M_{\text{th,sonic}} = 1$ , and Relation (II.4.4) transforms to:

$$R_{\text{pcr}} = \left( 1 + \frac{\gamma-1}{2} \right)^{-\frac{\gamma}{\gamma-1}}. \quad (\text{III.4.6})$$

Note that for air with  $\gamma = 1.4$ , the well-known result  $R_{\text{pcr}} \simeq 0.52828$  appears.

The critical pressure ratio  $R_{\text{pcr}}$  is then expressed at the throat through Relation (III.4.6). The pressure ratio  $R_p$  is then to be expressed at the throat as well and compared to  $R_{\text{pcr}}$ . Similarly the pressure ratio reads:

$$R_p = \frac{p_{\text{th}}}{p_0} = \frac{p_{\text{th}}}{p_{\text{th}} \left( 1 + \frac{\gamma-1}{2} M_{\text{th}}^2 \right)^{\frac{\gamma}{\gamma-1}}} = \left( 1 + \frac{\gamma-1}{2} M_{\text{th}}^2 \right)^{-\frac{\gamma}{\gamma-1}}. \quad (\text{III.4.7})$$

Under form (III.4.7), the pressure ratio requires knowledge of the solution state at the throat  $\mathbf{W}_{\text{th}}$ , under the assumption of subsonic flow.

However, at this level, only the subsonic solution state  $\mathbf{W}_{\text{L}}^*$  resulting from the HLLC solver is directly known. From this state, the  $\mathbf{W}_{\text{th}}$  state can be determined at the throat, but an iterative resolution is required, see Section III.2. Nevertheless, solution existence may fail for  $\mathbf{W}_{\text{th}}$ . This typically happens when the flow regime is sonic at the throat. When such situation appears,  $\mathbf{W}_{\text{th}}$  as well as  $R_p$  are unavailable. The absence of mathematical solution suggests a sonic regime. However, as will be seen in the following, it is possible to reach this conclusion by reformulating the inequality  $R_p > R_{\text{pcr}}$  in the  $\mathbf{W}_{\text{L}}^*$  state.

Indeed, after some algebraic manipulations, the combination of Relations (III.4.2), (III.4.6) and (III.4.7) results in:

$$R_p > R_{\text{pcr}} \Leftrightarrow M_{\text{th}} < 1 \rightarrow \text{subsonic}. \quad (\text{III.4.8})$$

This result appears obvious but will be useful in the following. It shows that comparing the pressure ratio  $R_p$  to the critical one  $R_{\text{pcr}}$  is equivalent to comparing the Mach number at the throat, obtained under the subsonic assumption, to unity. The  $\mathbf{W}_{\text{th}}$  state is still unknown at this point. However, Relation (III.4.8) can be reformulated in the  $\mathbf{W}_{\text{L}}^*$  state.

Indeed, let us come back to Relation (III.2.21), describing the conservation of mass, specific entropy, and specific total enthalpy between states  $\mathbf{W}_{\text{L}}^*$  and  $\mathbf{W}_{\text{th}}$ ,

$$\frac{A_L}{A_{th}} = \frac{M_{th}}{M_L^*} \left( \frac{1 + \frac{(\gamma-1)}{2} M_L^{*2}}{1 + \frac{(\gamma-1)}{2} M_{th}^2} \right)^{\frac{\gamma+1}{2(\gamma-1)}}. \quad (III.4.9)$$

This relation is valid for an isentropic flow. When sonic conditions are considered, Relation (III.4.9) reduces to (III.3.7). Relation (III.4.9) links the Mach number at the throat  $M_{th}$  in the  $\mathbf{W}_{th}$  state to the Mach number  $M_L^*$  in the  $\mathbf{W}_L^*$  state, and depends only on the geometric areas  $A_L / A_{th}$  and the isentropic exponent  $\gamma$ , those being necessarily positive.

Between those two states, the Mach number varies due to the isentropic acceleration or deceleration induced by the throat area. The transition from  $M_{th}$  to  $M_L^*$  (or the opposite) requires an iterative method. However, the flow direction is unchanged so the transition from  $M_{th}$  to  $M_L^*$  does not change the sign of inequality (III.4.8). Relation (III.4.8) can then be reformulated in the  $\mathbf{W}_L^*$  state through Relation (III.4.9):

$$\begin{aligned} M_{th} &< 1, \\ \downarrow_{(Eq.III.4.9)} \quad \downarrow_{(Eq.III.4.9)} & \\ M_L^* &< M_{L,sonic}^*. \end{aligned} \quad (III.4.10)$$

One must bear in mind that  $M_{L,sonic}^*$  represents the Mach number in the  $\mathbf{W}_{L,sonic}^*$  state when this state is meant to bring sonic conditions at the throat.  $M_{L,sonic}^*$  is then different from 1. Its value is obtained by solving Equation (III.3.7) with the help of an iterative method.

### Concluding remarks

The previous analysis shows that comparing the pressure ratio  $R_p$  to the critical one  $R_{per}$  is equivalent to comparing the Mach number at the throat, obtained under the subsonic assumption, to unity. Moreover, it is also identical to comparing the Mach numbers in the  $\mathbf{W}_L^*$  state:

$$R_p > R_{per} \Leftrightarrow M_{th} < 1 \Leftrightarrow M_L^* < M_{L,sonic}^* = M_{max} \rightarrow \text{subsonic}. \quad (III.4.11)$$

The last form is more convenient in the present context as it does not involve the  $\mathbf{W}_{th}$  solution state at the throat, that may be unavailable.  $M_L^* < M_{L,sonic}^* = M_{max}$  compares the Mach number  $M_L^*$ , resulting from the HLLC solver and the subsonic assumption, to the Mach number  $M_{L,sonic}^*$  in the  $\mathbf{W}_{L,sonic}^*$  state that would be required to bring sonic conditions at the throat. It represents the maximum Mach number allowed  $M_{L,sonic}^* = M_{max}$  in the  $\mathbf{W}_L^*$  state for the flow to be subsonic. Beyond  $M_{max}$  the flow at the throat is necessarily sonic.

An iterative method is nonetheless needed to find  $M_{L,sonic}^*$  through Relation (III.3.7). However, a simple analysis of function (III.3.7) shows that there always exists a solution in the interval  $M_{L,sonic}^* \in [0^+, 1]$ .

### III.4 Summary of the method

#### Solution states determination for flux computation

The overall method is summarized hereafter, for the computation of the solution state at the throat area.

##### a) Flow direction

The HLL solver is used between the two fluid states associated to the two rooms to estimate the flow velocity direction at the throat. Relation (III.4) is used and the second component  $U_{mom}^* = \rho u$  of the state vector  $\mathbf{U}^*$  provides the sign of the velocity.

##### b) Subsonic assumption and critical state

An unchoked regime is supposed at the throat. The situation is depicted in Figure 6 representing a subsonic Riemann problem. However, as only situations involving  $A_L = A_R$  (see Section II) are considered, the HLLC solver of Toro et al. (1994) is used directly and provides the solution states  $\mathbf{W}_L^*$  and  $\mathbf{W}_R^*$ . The Mach numbers  $M_L^*$  and  $M_R^*$  are consequently known. According to the flow direction, one of these two states is retained for the computation of the fictitious critical area  $A_{cr}$ . More precisely, it is determined with the help of Relation (III.4.3).  $M_{L,sonic}^*$  is also computed by solving Relation (III.3.7) with the help of an iterative method. It represents the maximum Mach number  $M_{L,sonic}^* = M_{max}$  allowed in the  $\mathbf{W}_L^*$  state (alternatively in the  $\mathbf{W}_R^*$  state) for the flow to be subsonic at the throat. Beyond  $M_{max}$  the flow at the throat is necessarily sonic. Two situations may then occur.

##### i. Case 1: The throat area is larger than the critical area and the Mach number is less than $M_{max}$ , i.e., $A_{th} > A_{cr}$ and $M_L^* < M_{L,sonic}^* = M_{max}$

The solution resulting from the assumption of an unchoked flow at the throat is in agreement with the criteria describing subsonic conditions. The solution state,  $\mathbf{W}_L^*$  or  $\mathbf{W}_R^*$ , resulting from the HLLC solver is consequently valid. From this solution state, the solution at the throat  $\mathbf{W}_{th}$  is computed through the set of isentropic relations, i.e., (III.2.21) requiring an iterative method and (III.2.20), (III.2.18), (III.2.15). Solution state  $\mathbf{W}_{th}$  being known, the criteria  $R_p > R_{per}$  and  $M_{th} < 1$ , previously replaced by  $M_L^* < M_{L,sonic}^* = M_{max}$  are examined or checked.

##### ii. Case 2: At least one of the inequalities $A_{th} > A_{cr}$ and $M_L^* < M_{L,sonic}^* = M_{max}$ is not fulfilled

The subsonic HLLC solution is left out as it is in disagreement with the above criteria. Those indicate that the flow is choked at the throat. The subsonic Riemann problem of Figure 6 is replaced by the specific sonic Riemann problem depicted in Figure 7.



The actual Mach number in the  $\mathbf{W}_{L,sonic}^*$  or  $\mathbf{W}_{R,sonic}^*$  solution state is computed by solving Relation (III.3.7). The velocity in the same state is then obtained by solving Relation (III.3.9). Iterative procedures are necessary. The rest of the solution state is determined with Relations (III.3.1) and (III.3.2). Then the solution state at the throat  $\mathbf{W}_{th,sonic}$  is determined with Relations (III.3.10 – III.3.12).

### Flux computation and solution update

The solution, in terms of conservative variables  $\mathbf{U} = (\rho, \rho u, \rho E)^T$ , is updated with the help of the Godunov (1959) first-order scheme,

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{\Delta t}{\Omega_i} \sum_{j=1}^{N_{faces}} \Phi_{ij}^*, \quad (\text{III.4.1})$$

where  $n+1$  and  $n$  denote two consecutive time steps and superscript  $*$  denotes the Riemann problem solution. Index  $i$  represents the current numerical cell, and index  $j$  the direct neighbors of cell  $i$ .  $\Omega_i$  is the volume of cell  $i$  and  $ij$  denotes the faces separating cells  $i$  and  $j$ .

Obviously, higher-order extensions can be considered but add complexity. Recall that a simple, fast, robust and accurate method dealing with very coarse 3D meshes is desired. Note also that the Godunov scheme is stable under the conventional CFL condition.

$\Phi_{ij}$  regroups the product of the fluxes  $\mathbf{F} = (\rho u \mathbf{n}, \rho u u \mathbf{n} + p \mathbf{n}, (\rho E + p) u \mathbf{n})^T$  at the cell's face  $ij$  in the normal ( $\mathbf{n}$ ) direction and the appropriate area, as given by (III.4.2). Indeed, recall that in the present situation, cell's face represents a wall containing an opening (a throat) as depicted in Figure 5. The area of the reflective wall also contributes to the flux distribution. Indeed, the integration of the conservation laws (III.1) on a control volume containing the perforated wall involves the following effective flux:

$$\Phi^* = A_{fluid} \mathbf{F}_{th} + A_{wall} \begin{pmatrix} 0 \\ p_i \mathbf{n} \\ 0 \end{pmatrix}, \quad (\text{III.4.2})$$

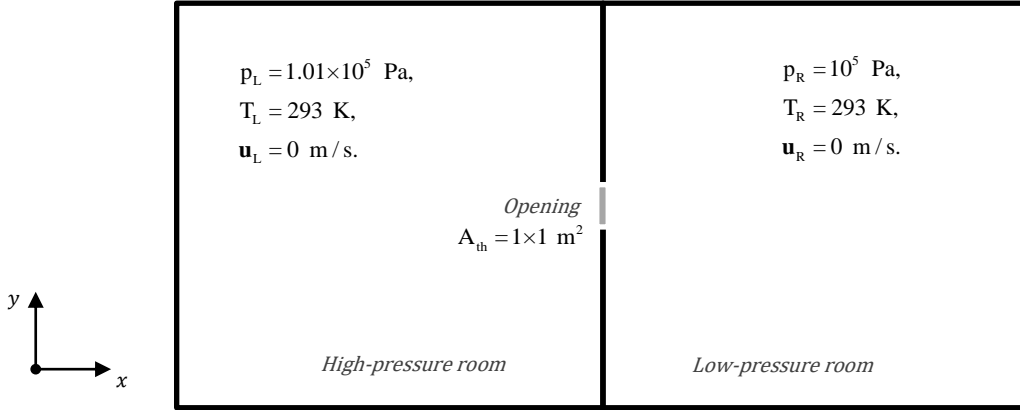
where  $\mathbf{F}_{th}$  contains the solution of the specific Riemann problem, either subsonic or sonic (summarized above) when a throat is present. When there is no opening on the cell's face, it reduces to the HLLC fluxes,  $\mathbf{F}_{th} = \mathbf{F}_{HLLC}$ .

$A_{fluid}$  represents the cross-section through which the fluid flows. When an opening is present, it corresponds to the throat area  $A_{fluid} = A_{th}$ . Otherwise, it is simply the cross-section of the cell's face:  $A_{fluid} = S_{ij}$ . Finally,  $A_{wall}$  represents the area of the wall, i.e., the cross-section of the cell's face  $S_{ij}$  without the fluid area:  $A_{wall} = S_{ij} - A_{fluid}$ .

## IV. Validations and illustrations

### IV.1 Simplified building

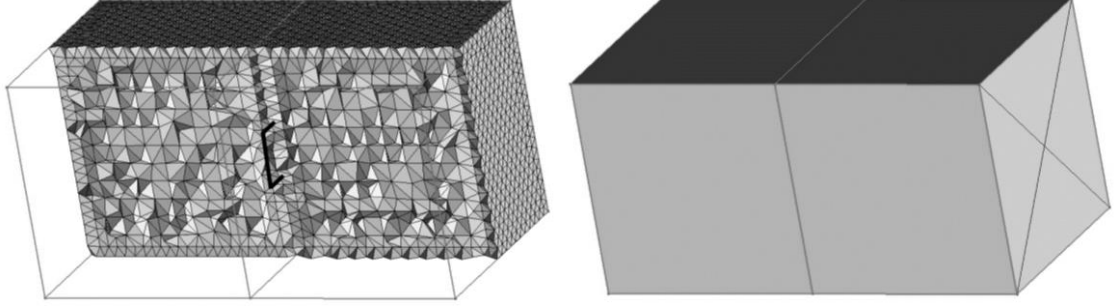
Computed results with the new method using very coarse meshes are compared with conventional 3D computations. A simplified building made of only two rooms is first considered with various pressure conditions and variable throat areas. Figure 8 displays the first configuration and associated initial conditions.



**Figure 8:** Schematic representation of a simple building made of two rooms only. The two rooms are of volume  $V = 4 \times 4 \times 4 \text{ m}^3$ . An opening separates the two rooms. Its area is  $A_{th} = 1 \times 1 \text{ m}^2$ . Air ( $\gamma = 1.4$ ) is initially at rest  $\mathbf{u}_L = \mathbf{u}_R = 0 \text{ m/s}$  and at temperature  $T_L = T_R = 293 \text{ K}$ . The initial pressure in the room on the left (high-pressure room) is  $p_L = 1.01 \times 10^5 \text{ Pa}$ . In the room on the right (low-pressure room), the initial pressure is  $p_R = 10^5 \text{ Pa}$ . Shock-tube type conditions are then set. The boundary surfaces are treated as reflective walls.

As two different initial pressures are set in the two rooms, the initial pressure profile in the whole building is discontinuous and of Heaviside type. Moreover, as an opening separates the two rooms, the cross-section profile in the whole building is also discontinuous. At the opening, the geometric non-conservative term  $\mathbf{p} \frac{\partial A}{\partial x}$  in System (III.1) consists of the product of Heaviside and Dirac functions. We will see in the following that such severe conditions are well-handled by the present Riemann solver and flux distribution, addressing under-resolved 3D computations.

Results provided by the under-resolved 3D computations are compared to those provided by the conventional 3D computations. Such conventional computations involve fine meshes including every geometric detail such as doors and windows. As seen in Section II.2, the under-resolved computations involve few numerical elements and a special treatment to address the various openings. The two meshes dealing with the present test case (Figure 8) are provided in Figure 9.



**Figure 9:** 3D meshes used for the test case depicted in Figure 8. On the left, the mesh for the conventional 3D computation is partly shown. It consists of 61.270 tetrahedral elements. The opening is fully drawn and meshed. It is represented by the dark lines that have been purposely thickened for the sake of clarity. On the right, the mesh used for the under-resolved 3D computation is shown. Only 4 prismatic elements per room are used. The mesh is made from a linear extrusion needing only the “footprints” of the building.

The results are provided in terms of mean pressure and mean density in the two rooms, for the two computations. The mean density is numerically approximated as,

$$\bar{\rho} = \frac{1}{V} \int_V \rho dV \simeq \frac{1}{V} \sum_{i=1}^N \rho_i \Omega_i,$$

where  $V$  represents the volume of the corresponding room and  $\Omega_i$  the volume of every element  $i$  (out of  $N$ ) composing the room. The mean pressure is determined with the help of the mean density  $\bar{\rho}$ , the mean momentum components  $\bar{\rho} \bar{u}_{x,y,z}$ , and the mean total energy  $\bar{\rho} \bar{E}$ ,

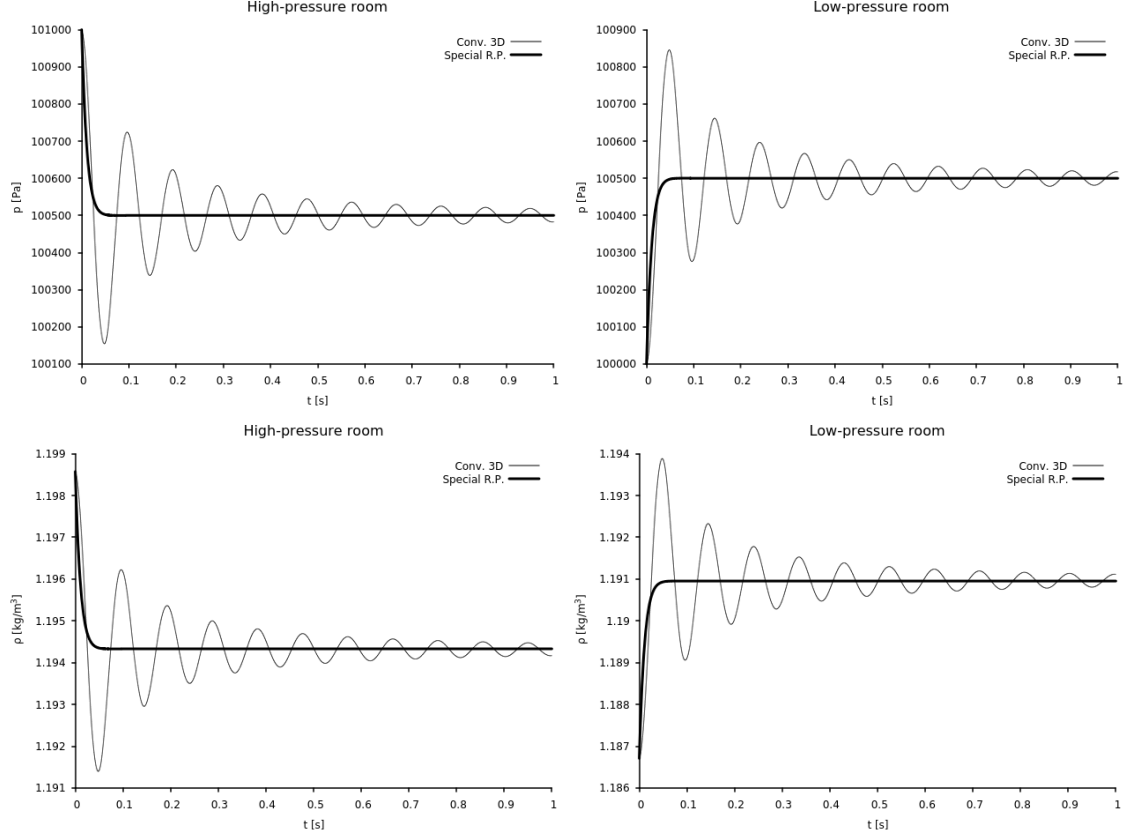
$$\bar{\rho} \bar{u}_{x,y,z} = \frac{1}{V} \int_V (\rho u_{x,y,z}) dV \simeq \frac{1}{V} \sum_{i=1}^N (\rho u_{x,y,z})_i \Omega_i,$$

$$\bar{\rho} \bar{E} = \frac{1}{V} \int_V (\rho E) dV \simeq \frac{1}{V} \sum_{i=1}^N (\rho E)_i \Omega_i.$$

The mean internal energy  $\bar{e}$  is then computed as,

$$\bar{e} = \bar{E} - \frac{1}{2} (\bar{u}_x^2 + \bar{u}_y^2 + \bar{u}_z^2),$$

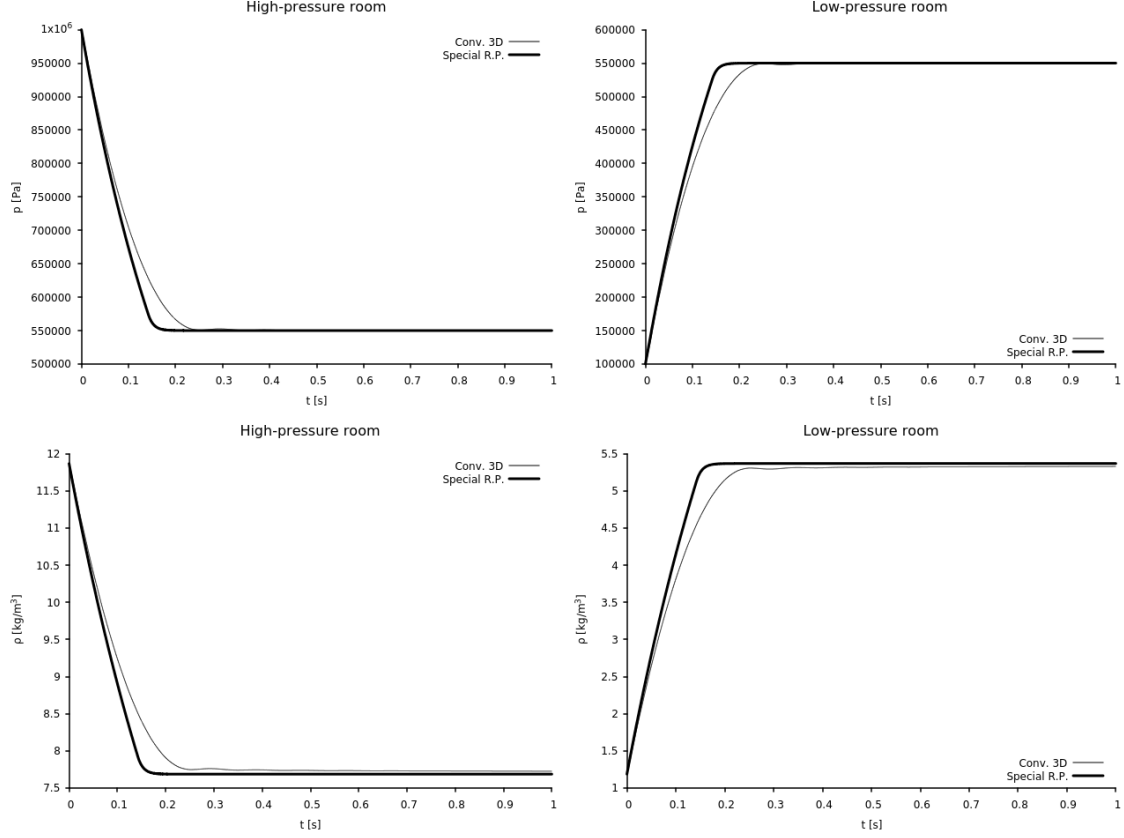
and the equation of state  $\bar{p}(\bar{\rho}, \bar{e})$  finally provides the mean pressure in the corresponding room. Results for the test case depicted in Figure 8 are shown in Figure 10.



**Figure 10:** Under-resolved 3D computation versus conventional 3D computation. The test configuration is depicted in Figure 8. It consists of a simple building made of two rooms separated by an opening. The rooms are of volume  $V = 4 \times 4 \times 4 \text{ m}^3$ . The opening is of area  $A_{th} = 1 \times 1 \text{ m}^2$ . The conventional 3D computation (denoted “Conv. 3D”) is performed on a mesh composed of 61.270 tetrahedral elements. The under-resolved computation (denoted “special R.P.” for special Riemann problem) is performed on a mesh made of 8 prismatic elements. The Godunov first-order scheme (III.4.1) is used with  $CFL = 0.5$ . The initial high pressure is  $1.01 \times 10^5 \text{ Pa}$ .

The initial pressure in the donor chamber has been taken weak ( $1.01 \times 10^5 \text{ Pa}$ ). In such conditions, only the subsonic part of the present Riemann solver is called. The mean density and mean pressure fields, important for many applications, are well-determined with the under-resolved computation that required less than 1 second with a sequential implementation. The conventional computation naturally shows more details of the wave dynamic solution. However, it required 2 hours and 29 minutes with a sequential implementation as well.

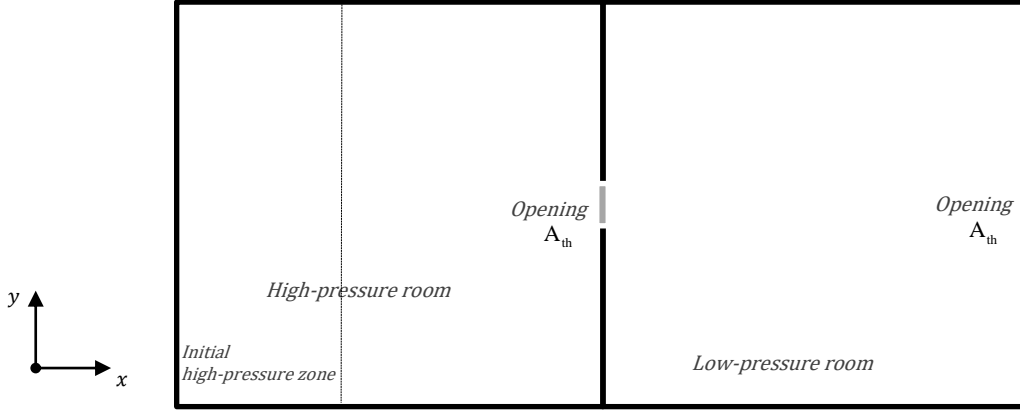
The test is now repeated with a high pressure of  $p_L = 10^6 \text{ Pa}$ . The pressure being higher, a sonic situation occurs in the early stages of the solution. The corresponding results are provided in Figure 11.



**Figure 11:** Under-resolved 3D computation versus conventional 3D computation. The test configuration of Figure 10 is repeated with a high pressure of  $p_L = 10^6$  Pa yielding choking conditions at early times.

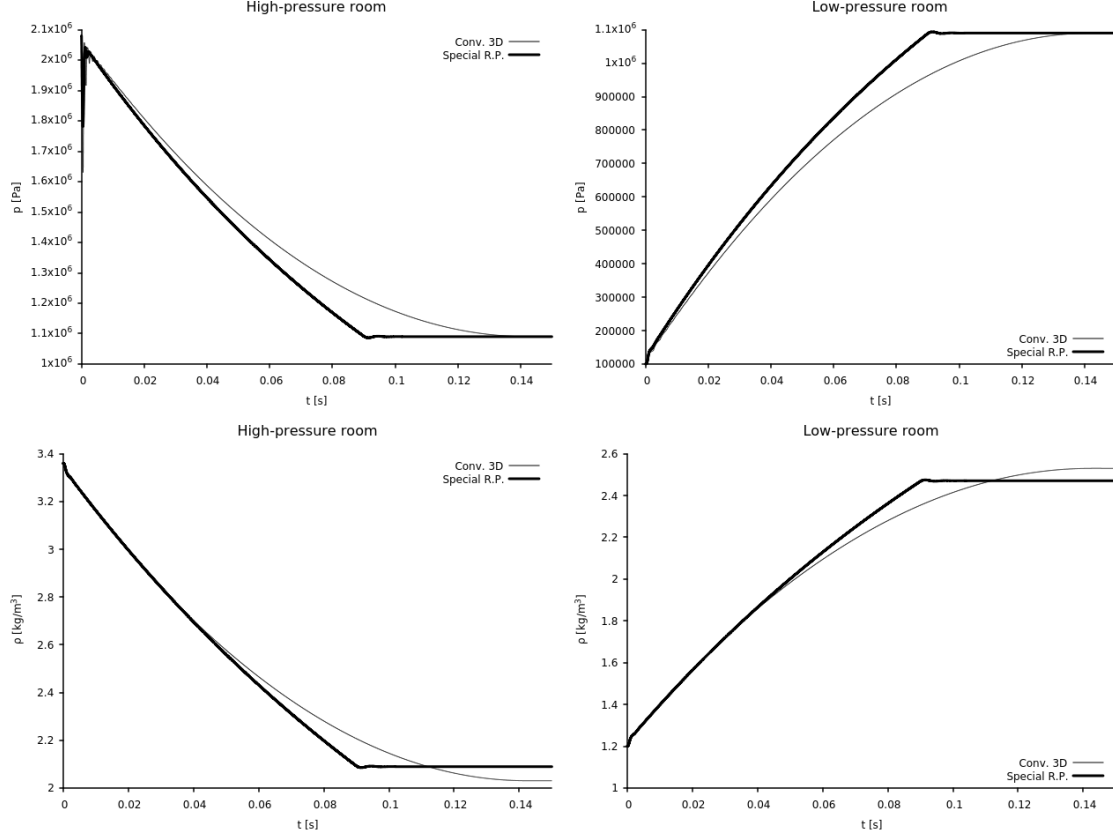
Before the pressure reaches its equilibrium value (quasistatic pressure), sonic flow occurs through the opening. The results provided by the under-resolved computation are again in good agreement with those obtained by the conventional computation, both in terms of pressure relaxation time, and mean (quasistatic) pressure field. The density profile agrees with the solution of the conventional computation as well.

To test the method further, the previous configuration is now slightly modified. Figure 12 displays the configuration and initial conditions.



**Figure 12:** Schematic representation of a simple building made of only two rooms. The two rooms are of volume  $V = 1 \times 1 \times 1 \text{ m}^3$ . A high pressure is set in a portion of the high-pressure room. This zone is of dimensions  $0.2 \times 1 \times 1$ . An opening separates the two rooms, and a second opening separates the low-pressure room and the exterior. Air is initially at rest. The boundary surfaces are treated as reflective walls, except for the opening connected to the atmosphere. The Appendix provides details for the treatment of boundary conditions.

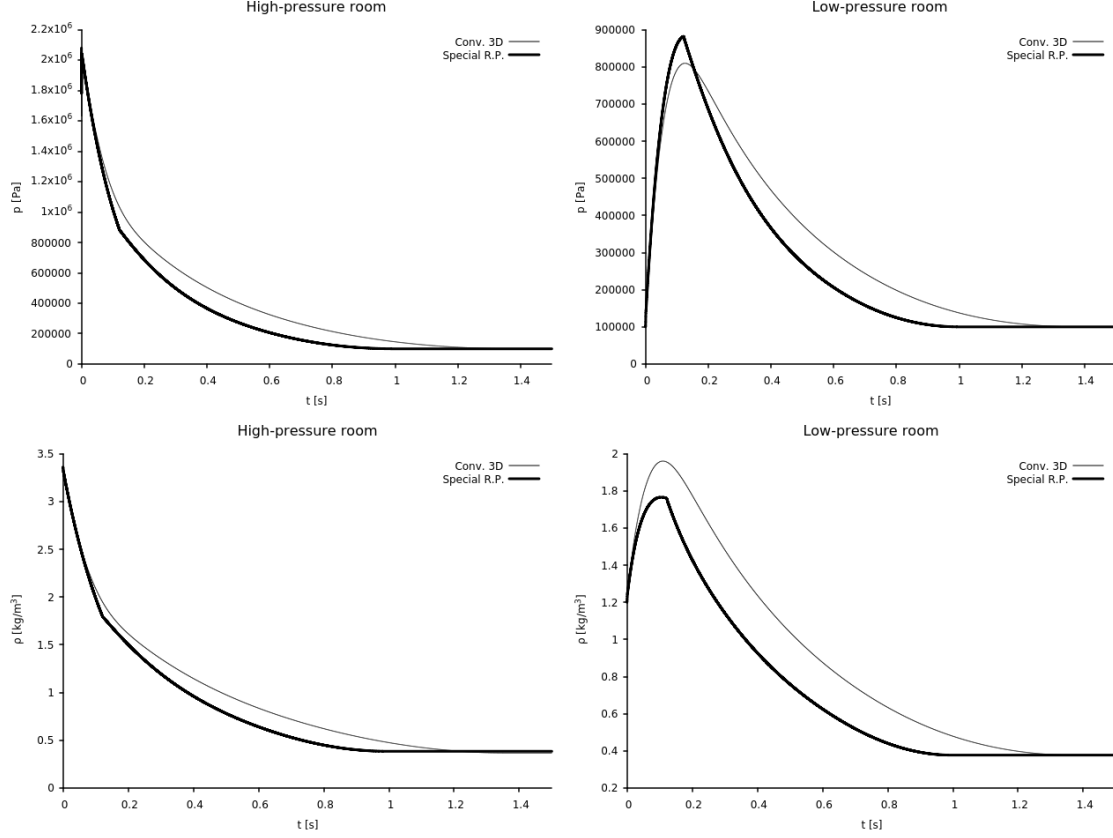
The high pressure is now initially set in only a portion of the high-pressure room. Besides a second opening is considered. This additional opening separates the building from the exterior. In the following test, the high pressure is increased up to  $10^7 \text{ Pa}$ , and the area of the opening separating the two rooms is lowered to  $A_{th} = 1 \times 0.01 \text{ m}^2$ , creating arduous sonic and subsonic conditions as time evolves. Results are provided in Figure 13 in terms of mean pressure and mean density both in the high- and low-pressure rooms. First, the opening separating the building from the atmosphere is considered closed and reflective wall conditions are used. It will be considered open (connected to the atmosphere) for the next test configurations.



**Figure 13:** Under-resolved 3D computation versus conventional 3D computation. The test configuration is depicted in Figure 12. It consists of a simple building made of two rooms separated by an opening. The rooms are of volume  $V=1\times1\times1\text{ m}^3$ . The opening is of area  $A_{th}=1\times0.01\text{ m}^2$ . The high pressure is initially set in only a portion of the high-pressure room. The initial conditions are  $p_{HP}=10^7\text{ Pa}$ ,  $\rho_{HP}=12\text{ kg/m}^3$  and  $\mathbf{u}_{HP}=0\text{ m/s}$ . In the rest of the building, the initial conditions are  $p_{LP}=10^5\text{ Pa}$ ,  $\rho_{LP}=1.2\text{ kg/m}^3$  and  $\mathbf{u}_{LP}=0\text{ m/s}$ . The conventional 3D computation (denoted “Conv. 3D”) is performed on a mesh composed of 66.565 tetrahedral elements. The under-resolved computation (denoted “special R.P.” for special Riemann problem) is performed on a mesh made of 12 prismatic elements. The Godunov first-order scheme (III.4.1) is used with  $CFL=0.5$ .

Good agreement between the two computations appears, both in terms of pressure relaxation time and mean pressure field. The density solution also agrees with the solution provided by the conventional computation.

The previous test is now repeated with the second opening separating the low-pressure room from the exterior, now connected to the atmosphere. Appropriate treatment of such boundary is reported in the Appendix. Its area is the same as the area of the opening separating the two rooms  $A_{th}=1\times0.01\text{ m}^2$ . Initial conditions remain the same. Atmospheric conditions are  $p_{atm}=10^5\text{ Pa}$ , and  $T_{atm}=289.75\text{ K}$ . Results are shown in Figure 14.

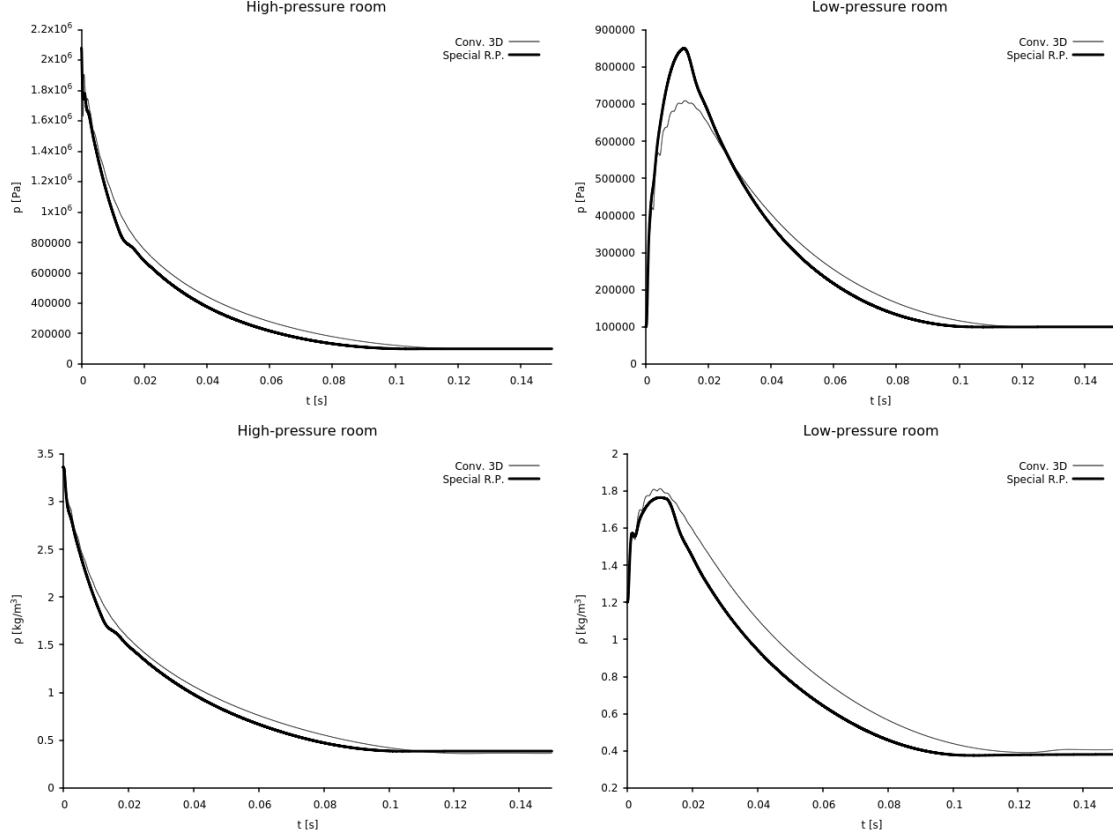


**Figure 14:** Under-resolved 3D computation versus conventional 3D computation. The test configuration of Figure 13 is repeated with a second opening of area  $A_{th} = 1 \times 0.01 \text{ m}^2$ . This opening separates the low-pressure room to the exterior where the atmospheric conditions dwell,  $p_{atm} = 10^5 \text{ Pa}$ , and  $T_{atm} = 289.75 \text{ K}$ . The conventional 3D computation (denoted “Conv. 3D”) is performed on a mesh composed of 71.309 tetrahedral elements. The under-resolved computation (denoted “special R.P.” for special Riemann problem) is performed on a mesh made of 12 prismatic elements.

Good agreement between the two computations appears one more time. Relaxation to the atmospheric conditions, induced by the window, is clearly seen and the solution agrees with the results of the conventional computation.

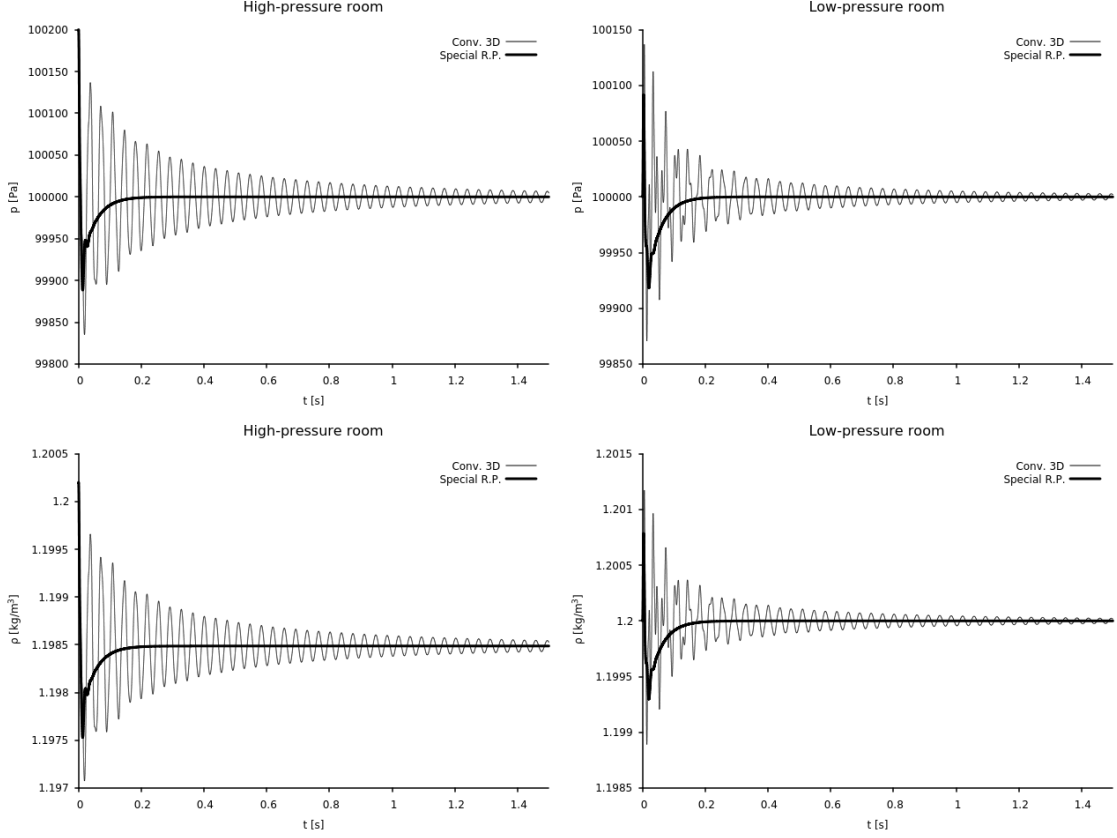
In the following test, areas of both openings are increased up to  $A_{th} = 1 \times 0.1 \text{ m}^2$ . Results are shown in Figure 15.





**Figure 15:** Under-resolved 3D computation versus conventional 3D computation. The test configuration of Figure 14 is repeated with an area of  $A_{th} = 1 \times 0.1 \text{ m}^2$  for the two openings. The conventional 3D computation (denoted “Conv. 3D”) is performed on a mesh composed of 60.602 tetrahedral elements. The under-resolved computation (denoted “special R.P.” for special Riemann problem) is performed on a mesh made of 12 prismatic elements.

The area of the two openings is 10 times greater than previously. Again, good agreement in terms of relaxation time and mean pressure and mean density fields, between the two computations, is observed. In the following test, the high-pressure in the donor room is lowered to  $p_{HP} = 1.01 \times 10^5 \text{ Pa}$ . Results are shown in Figure 16.



**Figure 16:** Under-resolved 3D computation versus conventional 3D computation. The test configuration of Figure 15 is repeated with an initial high pressure of  $p_{HP} = 1.01 \times 10^5$  Pa, and an initial high density of  $\rho_{HP} = 1.201$  kg/m<sup>3</sup>.

Similarly to Figure 10, where the high pressure is quite low, the conventional computation shows more details of the wave dynamic solution than the under-resolved computation. However, the mean density and mean (quasistatic) pressure fields are one-more time well-determined with the under-resolved computation.

Various levels of pressure and area have been considered to address a simple building made of two rooms. The present treatment of the Riemann problem involving cross-section variations yields results in good agreement with those provided by a conventional computation, both in terms of pressure relaxation time and quasistatic pressure field.

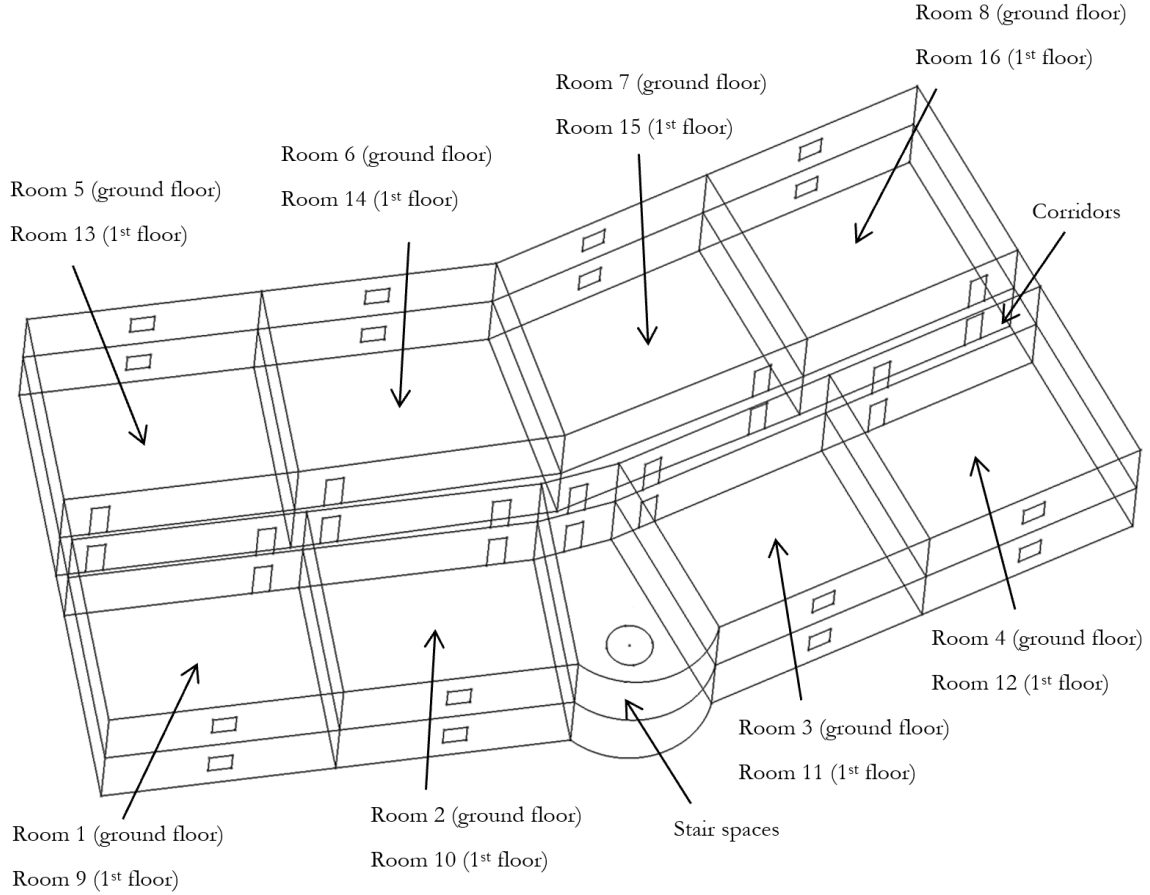
The gain in CPU time is tremendous due to the simplified mesh construction involving few numerical elements. The reported CPU times are provided in Table 1.

	Conventional computation	Under-resolved computation
	CPU time (sequential)	CPU time (sequential)
<b>Figure 10</b>	2 hours and 29 minutes	< 1 second
<b>Figure 11</b>	2 hours and 42 minutes	< 1 second
<b>Figure 13</b>	20 hours and 23 minutes	< 1 second
<b>Figure 14</b>	134 hours and 25 minutes	5 seconds
<b>Figure 15</b>	5 hours and 29 minutes	< 1 second
<b>Figure 16</b>	24 hours and 16 minutes	2 seconds

**Table 1:** CPU time reported for the various tests addressing simple geometries. The computations are performed with a sequential implementation.

## IV.2 Realistic building

To illustrate the capabilities of the method, the 3D building of Figure 4 is now addressed, in the direction of flow computations in realistic buildings. The conventional computation uses a 3D mesh made of about 1 million tetrahedral elements. The under-resolved computation uses a 3D mesh made of 116 prismatic elements. The building involves 35 openings composed of 18 doors of area  $0.75 \times 2 \text{ m}^2$ , 16 windows of area  $1 \times 1 \text{ m}^2$  and 1 staircase hole of area  $3 \text{ m}^2$ . The full 3D geometry is depicted in Figure 17.

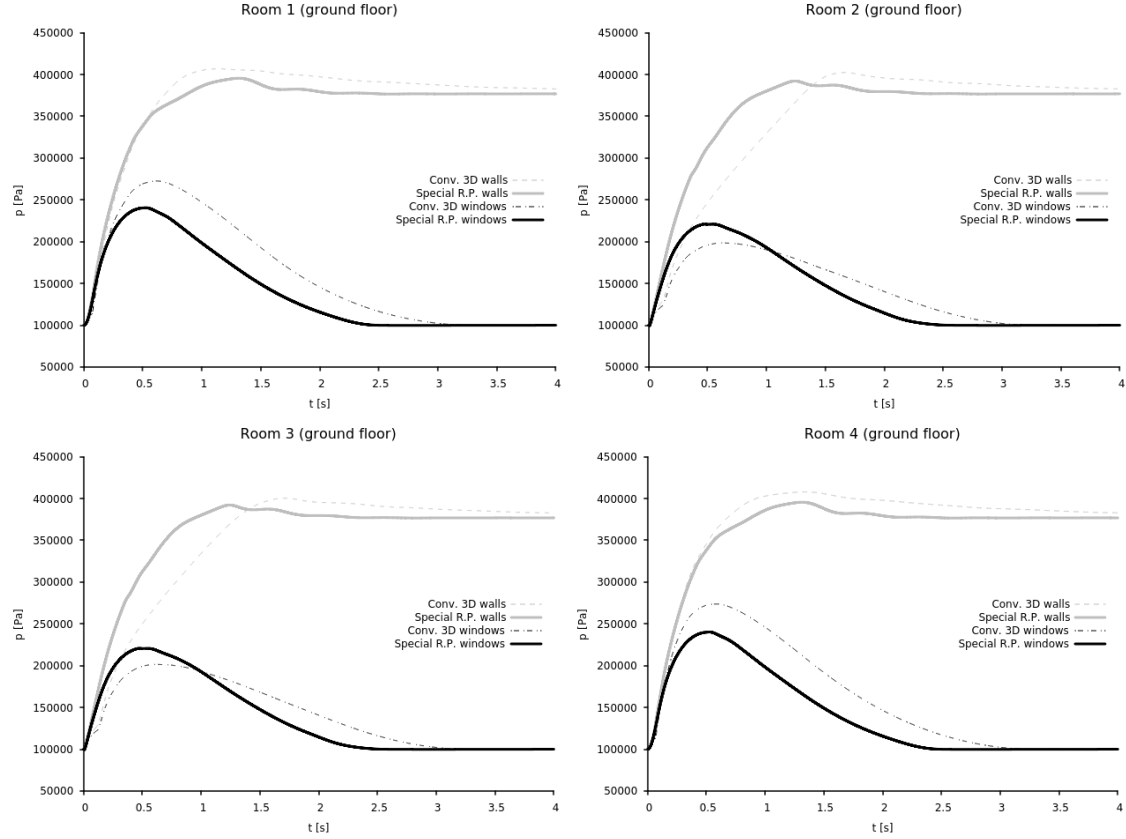


**Figure 17:** Full 3D geometry of a realistic building made of 16 rooms, 2 stair spaces and 2 corridors. The building involves 35 openings composed of 18 doors of area  $1.5 \text{ m}^2$ , 16 windows of area  $1 \text{ m}^2$  and 1 staircase hole of area  $3 \text{ m}^2$ . The stair space on the ground floor (hall) is initially set with a high pressure (HP) of  $p_{\text{HP}} = 10^7 \text{ Pa}$ . The pressure in the rest of the building is  $p = 10^5 \text{ Pa}$ . Air is initially at rest  $\mathbf{u} = 0 \text{ m/s}$ , and at temperature  $T = 293 \text{ K}$  in the whole building. Atmospheric conditions are  $p_{\text{atm}} = 10^5 \text{ Pa}$ , and  $T_{\text{atm}} = 289.75 \text{ K}$ . The building is meshed with about 1 million tetrahedral elements. Computation on the full geometry is to be compared to the under-resolved computation where the openings are not meshed, and few elements are used (see Figure 4).

Results in terms of mean pressure are provided in the following figures. Mean densities are not presented for the sake of space. Two sets of computation are carried out. The first one considers all windows as closed. The corresponding surfaces are then treated as reflective walls. The second considers all windows as open and connected to the atmosphere. Note that the purpose of the present tests is to compare results from under-resolved

computations to results provided by conventional computations. More realistic situations, where inner walls get gradually destroyed under the effects of pressure are part of future investigations.

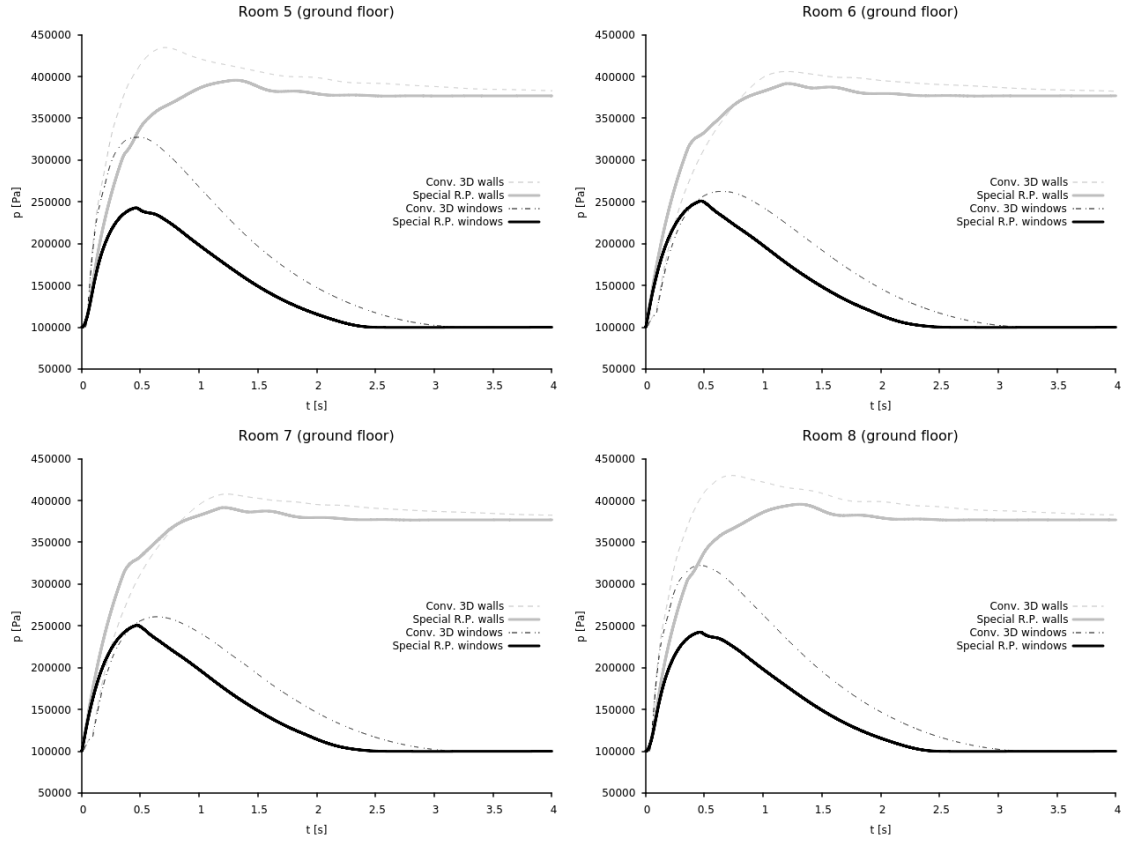
Rooms on the ground floor are first addressed. Figure 18 shows the results for the rooms on the south side of the building shown in Figure 17.



**Figure 18:** Under-resolved 3D computation versus conventional 3D computation. The test configuration is depicted in Figures 4 and 17. It consists of a realistic building made of two floors. Each floor is made of 8 rooms with doors and windows, 1 stair space and 1 corridor. In the hall on the ground floor (stair space, see Figure 17), the pressure  $p_{HP} = 10^7$  Pa is initially set. In the rest of the building, the initial pressure is  $p = 10^5$  Pa. Air is initially at rest  $\mathbf{u} = 0$  m/s, and a temperature  $T = 293$  K in the whole building. Atmospheric conditions are  $p_{atm} = 10^5$  Pa, and  $T_{atm} = 289.75$  K. The conventional 3D computation (denoted “Conv. 3D”) is performed on a mesh composed of about 1 million tetrahedral elements. The under-resolved computation (denoted “special R.P.” for special Riemann problem) is performed on a mesh made of 116 prismatic elements. The Godunov first-order scheme (III.4.1) is used with  $CFL = 0.5$ . The mean pressure is plotted both for the closed-window situation (denoted as “walls”) and the open-window situation (denoted as “windows”). The present figure shows the results for the four rooms located on the ground floor, on the south side of the building (see Figure 17).

Results provided by the under-resolved computation show a very reasonable agreement with those provided by the conventional computation, both for the situation where all windows are closed (reflective walls) and for the situation where all windows are open. In this latter situation, atmospheric conditions significantly affect the solution.

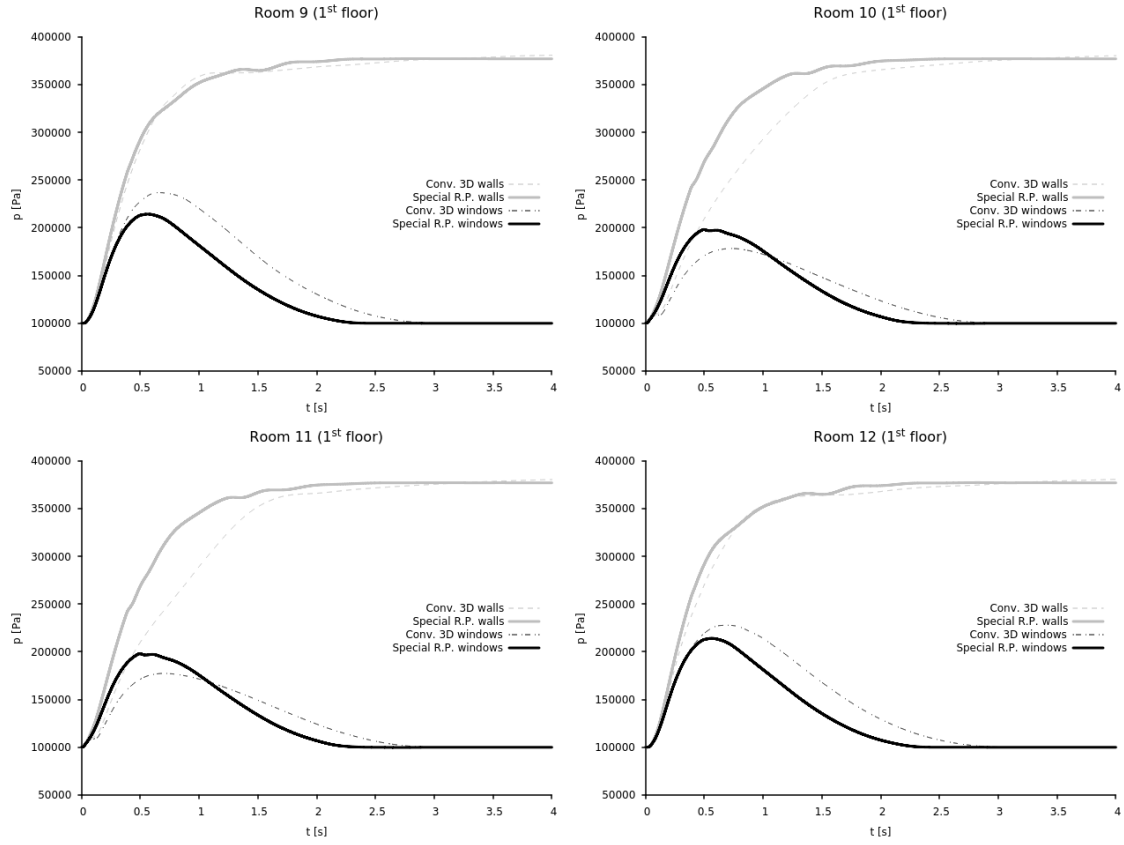
Figure 19 shows the results for the rooms on the north side of the building (see Figure 17).



**Figure 19:** Under-resolved 3D computation versus conventional 3D computation. The test configuration consists of the realistic building depicted in Figures 4 and 17. The present figure shows the results for the four rooms located on the ground floor, on the north side of the building (see Figure 17).

Results provided by the under-resolved computations appear quite reasonable. The simplicity and rapidity of the method yield sometimes less accurate pressure spikes as seen in the plots for rooms 5 and 8. However, the overall pressure spikes remain reasonably estimated.

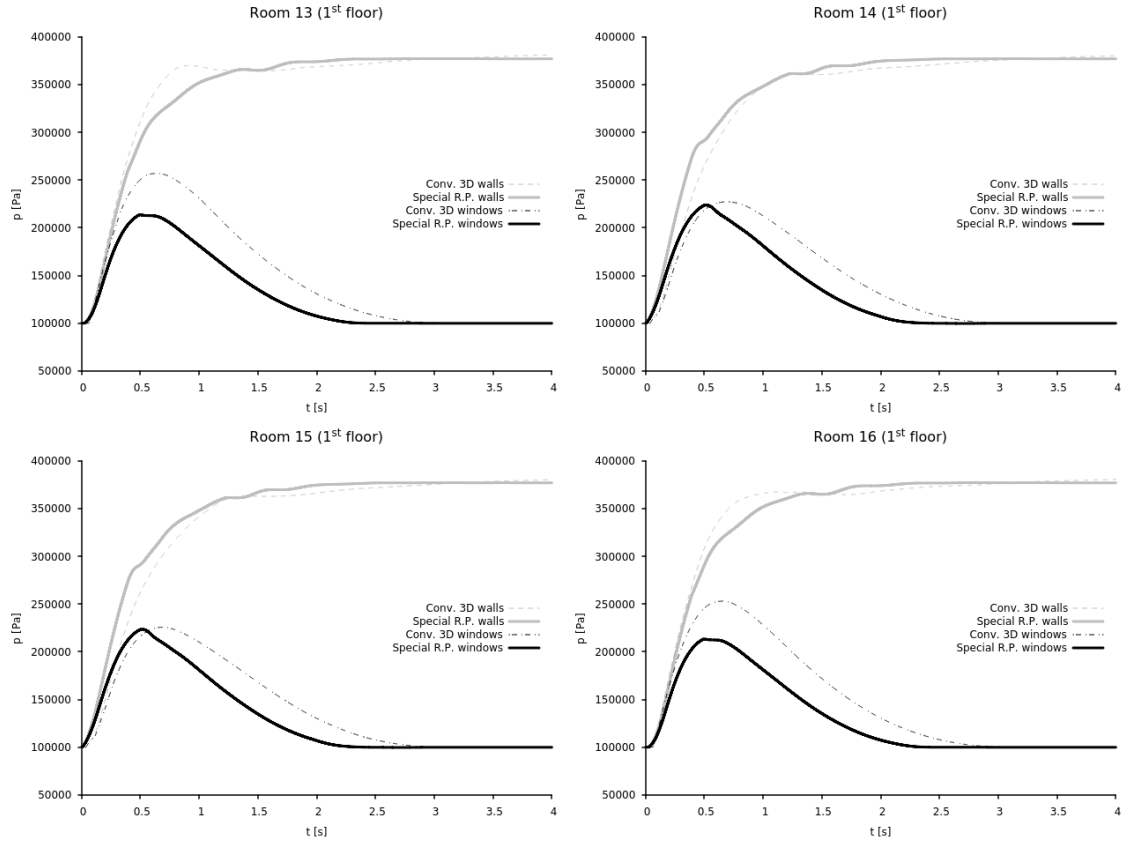
Rooms on the first floor are now addressed. Figure 20 shows the results for the rooms on the south side of the building (see Figure 17).



**Figure 20:** Under-resolved 3D computation versus conventional 3D computation. The test configuration consists of the realistic building depicted in Figures 4 and 17. The present figure shows the results for the four rooms located on the first floor, on the south side of the building (see Figure 17).

Again, a very reasonable agreement is observed between the conventional and under-resolved computations, for both situations where the windows are closed (reflective walls) or open (connected to the atmosphere).

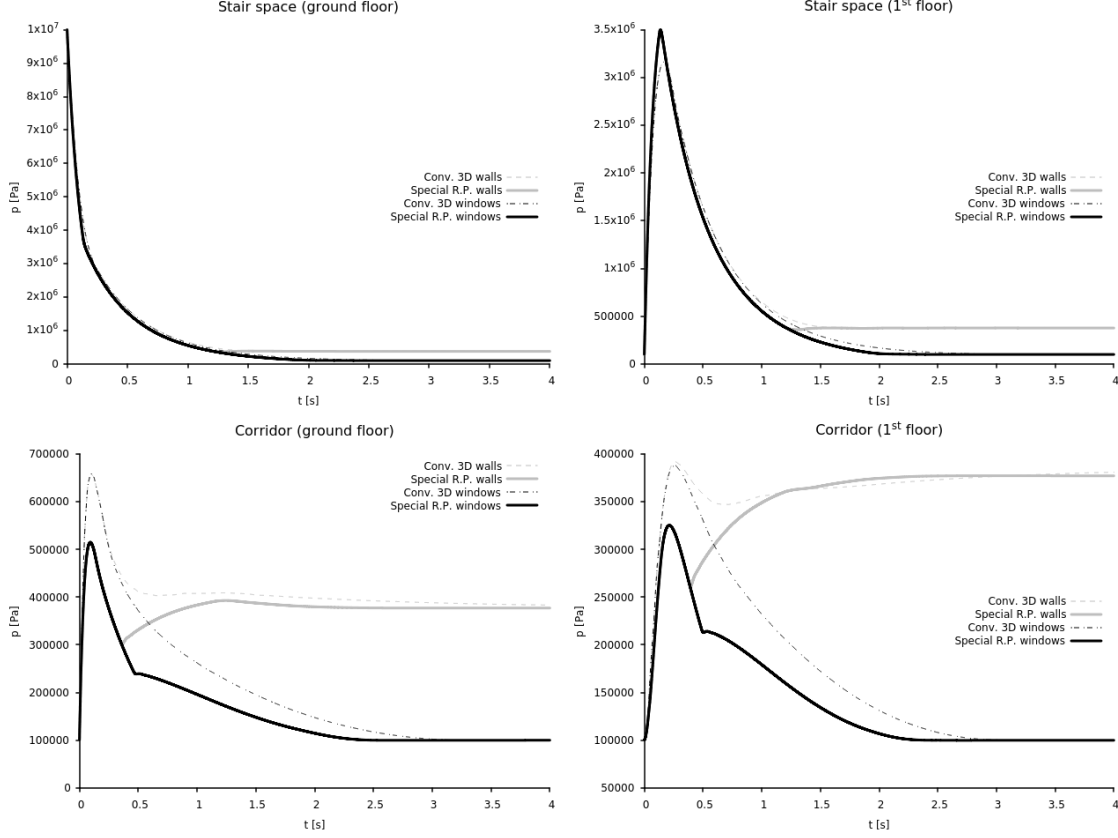
Figure 21 shows the results for the rooms on the north side of the building (see Figure 17).



**Figure 21:** Under-resolved 3D computation versus conventional 3D computation. The test configuration consists of the realistic building depicted in Figures 4 and 17. The present figure shows the results for the four rooms located on the first floor, on the north side of the building (see Figure 17).

One more time, a quite reasonable agreement is observed. As in Figure 19, less accurate pressure spikes appear in the plots for rooms 13 and 16. Nevertheless, the overall pressure spikes remain reasonably estimated.

Finally, stair spaces and corridors are addressed. Results are provided in Figure 22.



**Figure 22:** Under-resolved 3D computation versus conventional 3D computation. The test configuration consists of the realistic building depicted in Figures 4 and 17. The present figure shows the results for the stair spaces and the corridors (ground and first floors, see Figure 17). The initial high-pressure zone  $p_{HP} = 10^7$  Pa, is in the stair space, on the ground floor.

The high pressure is initially set in the stair space (hall), on the ground floor. Pressure fields in the stair spaces are in good agreement with the pressure profiles provided by the conventional computations. Results in the corridors are quite reasonable as well.

The simplicity and rapidity of the method yield sometimes less accurate pressure spikes. However, the overall results provided by the under-resolved computations show a very reasonable agreement with those provided by the conventional computations, both in terms of pressure relaxation time and mean pressure, such mean (or quasistatic) pressure field being of importance for many situations.

Compared to conventional computation, construction of the geometry is very simple and fast as only the 2D “footprints” of the building are necessary. A conforming mesh with as few elements and human labor as possible is then constructed by extruding along the third dimension (see Section II). Geometric openings such as doors, windows, stairs are not meshed. The present Riemann problem takes into account the geometric details and a specific flux distribution is used (see Section III).

The overall method makes a fast numerical framework, both on the preprocessing stage and on the solver side. The agreement with results provided by the conventional computations is quite reasonable and the CPU time savings is tremendous as seen in Table 2.



	Conventional computation	Under-resolved computation
	CPU time (domain decomposition-MPI architecture using 63 CPUs)	CPU time (sequential)
Realistic building (walls)	4 hours, 46 minutes	8 seconds
Realistic building (windows)	4 hours, 18 minutes	8 seconds

**Table 2:** CPU time reported for the computations of the realistic building. The conventional computations are performed with a parallel implementation using MPI architecture and 63 CPUs. The under-resolved computations are performed with a sequential implementation due to the few numbers of numerical elements.

The previous conventional computations were rerun with coarse meshes. Because of the presence of doors and windows in the geometry, 4625 tetrahedral elements were necessary. More elements than the under-resolved computation (116 prismatic elements) were then used and the computation time is quite larger than the under-resolved computation's. The conventional computations required about 12 minutes with a sequential implementation. Results appear somewhat closer to the reference results provided by the conventional computations using a fine mesh (about 1 million elements). The corresponding results are not presented for the sake of space.

The CPU time is reduced to about 34 seconds when a parallel architecture is used with 35 CPUs. Much more computational resources are then required in comparison to under-resolved computations, such IT-resource dependence being undesirable under pressing circumstances.

Furthermore, because every geometric detail, corresponding to stairs, doors and windows in the present example, must be drawn and meshed, the geometry-construction-and-mesh-generation step requires much more time and human efforts than the proposed approach that is based on the 2D “footprints” of the geometry.

The proposed method, composed of simple and fast geometry-construction-and-mesh-generation and specific-Riemann-problem steps, then appears very helpful when hazardous and pressing situations are involved and require knowledge of the pressure fields.

Another important asset also rises. Indeed, because geometric restrictions are neither drawn nor meshed but are only marked for the specific Riemann solver, it appears straightforward to introduce a time-dependent throat area. Such situations may describe for instance the gradual destruction of a wall under the effect of pressure and will be investigated in future works. The treatment of such situation is not straightforward with the conventional approach where the openings are initially drawn and meshed, yielding consequently a mesh only adapted for the initial time.

## V. Conclusion

A fast numerical framework, both on the preprocessing stage and on the solver side, has been developed to address flow computations in complex geometries, such as buildings. Geometry and mesh constructions are very simple, fast, and flexible. The meshing tool used in this work is GMSH. The strategy consists of constructing only the “footprints” of the building geometry (nodes, lines, and surfaces), generate a coarse but conformal 2D mesh and then extrude along the third dimension. In this way, each room is discretized with few elements and human efforts.

The geometric details, such as doors, windows, staircases, are not required during the geometry-and-mesh-construction step. The preprocessing stage is consequently very fast, compared to conventional meshes' needing geometric details.

With the present approach, such geometric openings are treated through a specific Riemann solver that can handle both unchoked and choked situations occurring through the opening. Reflected waves are treated as well with the present Riemann solver.

Such Riemann solver is simple and robust. It is based on the following observation. In the limit case where the cross-sections on both sides of the geometric discontinuity are the same  $A_L = A_R$ , the geometric discontinuity becomes transparent (for an unchoked flow). Such geometrical property is easily satisfied from the previous geometry-and-mesh-construction step, where a conformal constrained Delaunay-type mesh is built from given input nodes which ensures  $A_R = A_L$  at all rooms' separations by construction.

An essential simplification appears as there is no need to consider 4 waves and 3 states in the unchoked Riemann problem. In this particular case it reduces to 3 waves and 2 states, as done usually with the Euler equations, without cross-section variation. The simple and robust HLLC solver can then be used and provides the solution state upstream from the geometric opening. Isentropic relations are used afterwards to select the flow regime appropriate to the flow conditions, i.e., subsonic or sonic, and provide the solution state at the opening (the throat). Finally, a specific but simple flux distribution is performed.

The proposed overall method has been tested on both simple 3D geometries, with various levels of pressure and opening area, and a realistic building. Results provided by the under-resolved computations on simple 3D geometries show a very good agreement with results from the conventional computations. Such good agreement is obtained both in terms of pressure relaxation time, a direct consequence of flux computation, and in terms of mean (or quasistatic) pressure field.

When realistic complex buildings are addressed, quasistatic pressure and relaxation time appear reasonably accurate, making the present method a simple and very fast numerical tool to address flows in complex buildings.

It is worth mentioning that using very coarse meshes may yield a loss of accuracy where curved geometries are considered. One way to remedy to this drawback is to use high-order meshes, see for instance Dobrzynski and Jannoun (2017). This topic is part of future investigations.

Depending on the complexity of the building, and the available computational resources, a conventional computation using a coarse mesh may provide the desired results with low CPU time. However, because geometric details such as doors for example are needed, the geometry-construction preprocessing stage demands much more time and human labor than the proposed approach that is based on the 2D "footprints" of the geometry. Such tedious mesh generation is not desirable when hazardous and pressing situations are involved and require knowledge of the pressure fields.

The present method then appears very convenient in such circumstances. Because of the simple preprocessing stage, the gain in terms of human efforts is huge compared to conventional computations. The Riemann solver dealing with geometric reductions is quite simple as well. The present under-resolved 3D computations provide reasonably accurate mean (or quasistatic) pressure fields in such complex buildings, with minimum computational resources and CPU time.

Another major asset is in favor of the present method. Indeed, because geometric restrictions are neither drawn nor meshed but are only marked for the specific Riemann solver, it appears straightforward to introduce a time-dependent throat area, unlike conventional computations. Such situations may describe for instance the gradual destruction of a wall under the effect of pressure and is also part future investigations.

## Acknowledgments

This work was partly funded by CEA Gramat. Antoine Osmont, Sébastien Courtiaud and Emmanuel Lapébie are gratefully acknowledged.

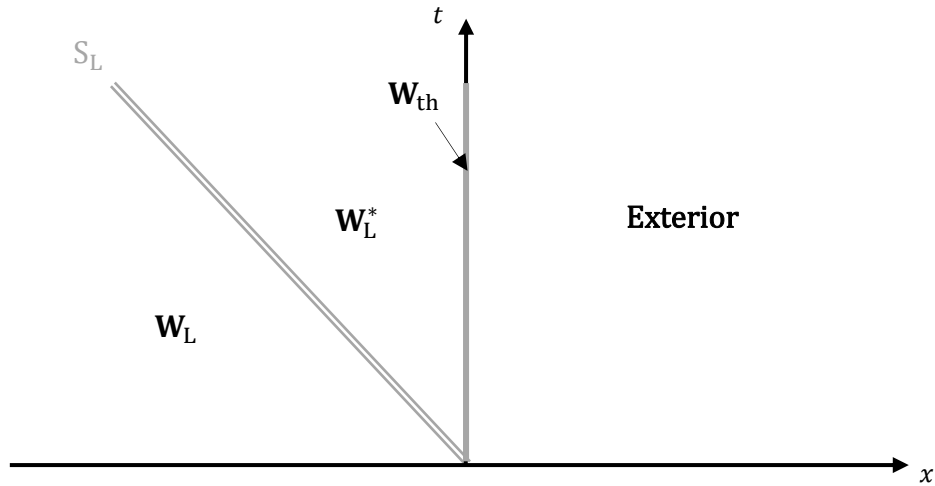
## Declaration of Competing Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Appendix. Boundary condition

The Riemann solver presented in Section III, taking into account the effects of a dimensional reduction, is adapted to a boundary cell's face. Such situation describes for example a flow occurring through a window, placed on a boundary wall of a building.

To illustrate the situation, let us once again use the flow situation depicted in Figure 6. However, as the cell's face separates the numerical domain from the exterior, the waves appearing on the right side of Figure 6 are now multidimensional. Yet, mathematical relations across such waves are unavailable. The states on the right of the opening in Figure 6 are then inaccessible and the Riemann problem situation transforms to the one depicted in Figure A.1, involving only one extreme wave and the stationary wave, whether the flow is subsonic or sonic at the throat.



**Figure A.1:** Schematic representation of the wave diagram of 1D Riemann problem for a boundary cell's face. A wall presenting a window (throat) separates a room on the left from the exterior. In the present example, the fluid flows from the left of the opening (numerical domain) to the right (exterior). The geometric discontinuity is indicated with the full line. Only one extreme wave propagates into the numerical domain.

The Riemann problem for a boundary cell's face is then specific in the sense that only two waves appear whether or not the flow is choked at the throat (window area). The method is presented hereafter based on the flow situation depicted in Figure A.1, involving a fluid flowing from the left of the opening (numerical domain) to the right (exterior). Naturally the method treats the reversed flow situation similarly.

### A.1 Sonic flow

When the flow is choked, the upstream side (numerical domain) is isolated from the downstream side (exterior) at the throat. In such conditions, the Riemann problem is the

same as the one presented in Section III.3. The corresponding sonic Riemann solver is then directly used and provides the solution state at the throat  $\mathbf{W}_{\text{th,sonic}}$ , as well as the  $\mathbf{W}_{\text{L,sonic}}^*$  solution state (or alternatively the  $\mathbf{W}_{\text{R,sonic}}^*$  if the flow is reversed). The solutions fluxes are computed according to the method presented in Section (III.4).

## A.2 Unchoked flow

However, when the flow is unchoked, the Riemann solver of Section III.2 must be adapted. Across the extreme wave, the acoustic approximation is one more time used to simplify the calculations. A relation linking the speed  $u_L^*$  to the pressure  $p_L^*$  appears:

$$u_L^*(p_L^*) = \frac{p_L - p_L^* + Z_L u_L}{Z_L}. \quad (\text{A.1})$$

Note that with the present example (Figure A.1), the fluid flows from the left of the opening to the right. The extreme wave  $S_L$  then propagates into the  $\mathbf{W}_L$  unperturbed state. In situations where the flow is reversed, Relation (A.1) transforms to,

$$u_R^*(p_R^*) = -\frac{p_R - p_R^* - Z_R u_R}{Z_R},$$

as sign “-” shall be used in the acoustic approximation.

Moreover, the linearized version of Laplace’s law, based on the sound speed definition, is used once again. A relation linking the density  $\rho_L^*$  to the pressure  $p_L^*$  appears:

$$\rho_L^*(p_L^*) = \rho_L + \frac{p_L - p_L^*}{c_L^2}. \quad (\text{A.2})$$

Finally, recall that between the state  $\mathbf{W}_L^*$  and the state at the throat  $\mathbf{W}_{\text{th}}$ , the flow is isentropic and stationary resulting in the following relations:

$$\rho_L^* u_L^* A_L^* = \rho_{\text{th}} u_{\text{th}} A_{\text{th}}, \quad (\text{A.3})$$

$$\frac{p_L^*}{\rho_L^{*\gamma}} = \frac{p_{\text{th}}}{\rho_{\text{th}}^\gamma}, \quad (\text{A.4})$$

$$\frac{\gamma p_L^*}{(\gamma-1)\rho_L^*} + \frac{1}{2} u_L^{*2} = \frac{\gamma p_{\text{th}}}{(\gamma-1)\rho_{\text{th}}} + \frac{1}{2} u_{\text{th}}^2. \quad (\text{A.5})$$

Laplace’s law (A.4) and mass conservation (A.3) yield:

$$\rho_{\text{th}}(p_L^*) = \rho_L^*(p_L^*) \left( \frac{p_{\text{th}}}{p_L^*} \right)^{\frac{1}{\gamma}}, \quad (\text{A.6})$$

$$u_{\text{th}}(p_L^*) = \frac{\rho^*(p_L^*) u^*(p_L^*) A_L}{\rho_{\text{th}}(p_L^*) A_{\text{th}}}. \quad (\text{A.7})$$

For a subsonic flow occurring through the throat, it is fairly conceivable to consider that the multidimensional waves, propagating into the exterior, quickly impose the exterior pressure  $p_{\text{ext}}$ . Such observation closes the mathematical system as the pressure at the throat is supposed to be equal to the exterior pressure:

$$p_{\text{th}} = p_{\text{ext}}. \quad (\text{A.8})$$

The total specific enthalpy equation (A.5) becomes consequently a function depending only on the pressure  $p_L^*$  in the  $\mathbf{W}_L^*$  state,

$$\frac{\gamma p_L^*}{(\gamma-1) \rho_L^*(p_L^*)} + \frac{1}{2} u_L^{*2}(p_L^*) = \frac{\gamma p_{\text{th}}}{(\gamma-1) \rho_{\text{th}}(p_L^*)} + \frac{1}{2} u_{\text{th}}^2(p_L^*). \quad (\text{A.9})$$

Solution of the boundary Riemann problem, in the subsonic case, is then obtained by solving Relation (A.9) with the help of an iterative method. The remaining solution variables are determined with the previous relations and the solution fluxes are computed according to the method presented in Section (II.4).

### A.3 Flow regime

We then have in hands two Riemann solvers, subsonic and sonic, for a boundary cell's face. As in Section III.3, selecting the solver appropriate to the flow conditions is done with the help of the critical state.

The proposed method consists of assuming a subsonic (unchoked) flow and assessing the relevance of this assumption by comparing the subsonic solution to critical conditions. The first step is then to solve Relation (A.9). Sometimes, there may be no mathematical solution, indicating that the flow is choked at the throat. In Section II.3, this situation could be circumvented by reformulating the critical pressure ratio  $R_{\text{per}}$  at the throat in the  $\mathbf{W}_L^*$  state and with the help of the HLLC solver providing necessarily a mathematical solution state  $\mathbf{W}_L^*$ , even fictitious. The situation is different for a boundary cell's face. Indeed, no HLLC-type solver is available. The flow is then considered as choked if Relation (A.9) presents no solution.

However, when a solution, even fictitious, is available for Relation (A.9), both solution states  $\mathbf{W}_L^*$  and  $\mathbf{W}_{\text{th}}$  are known. The subsonic assumption is relevant if the following criteria (Section III.4) are fulfilled,

$$\left\{ \begin{array}{l} A_{\text{th}} > A_{\text{cr}}, \\ R_p > R_{\text{per}}, \\ M_{\text{th}} < 1, \\ M_L^* < M_{L,\text{sonic}}^* \end{array} \right. \rightarrow \text{subsonic},$$

with:

$$A_{cr} = A_L M_L^* \left( \frac{1 + \frac{(\gamma-1)}{2} M_L^{2*}}{1 + \frac{(\gamma-1)}{2}} \right)^{-\frac{\gamma+1}{2(\gamma-1)}},$$

$$R_p = \left( 1 + \frac{\gamma-1}{2} M_{th}^2 \right)^{-\frac{\gamma}{\gamma-1}},$$

$$R_{pcr} = \left( 1 + \frac{\gamma-1}{2} \right)^{-\frac{\gamma}{\gamma-1}}.$$

$M_{L,sonic}^* = M_{max}$  is obtained by solving Relation (III.3.7) with the help of an iterative method. The equation is reminder hereafter:

$$\frac{A_L}{A_{th}} = \frac{1}{M_{L,sonic}^*} \left( \frac{2}{\gamma+1} \left( 1 + \frac{\gamma-1}{2} M_{L,sonic}^{*2} \right) \right)^{\frac{\gamma+1}{2(\gamma-1)}}.$$

## References

- Andrianov, N. and Warnecke, G. (2004) On the solution to the Riemann problem for the compressible duct flow. *Siam J. Appl. Math.* 64(3), 878-901
- Coulter, G. A., Bulmash, G. and Kingery, C. N. (1988) Simulation techniques for the prediction of blast from underground munitions storage facilities. Army Ballistic Research Lab, Aberdeen Proving Ground, USA
- Davis, S. (1988) Simplified second-order Godunov-type methods. *SIAM Journal on Scientific and Statistical Computing*, 9(3), 445-473
- Dijkstra, E.W. (1959) A note on two problems in connexion with graphs. *Numerische Mathematik*, 1, 269-271
- Dobrzynski, C. and Jannoun, G. (2017) High order mesh untangling for complex curved geometries. INRIA Research Report RR-9120
- Frank, R. and Hurley, J. (2005) Fast running model for arbitrary room airblast. *Int. Symp. Interaction of the Effects of Munitions with Structures*, Orlando, USA
- Geuzaine, C. and Remacle, J.-F. (2009) Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering* 79(11), pp. 1309-1331
- Godunov, S. (1959) A finite difference scheme for numerical computation of the discontinuous wave solutions of equations of fluid dynamics. *Math. Sb.* 47 271-306
- Han, E., Hantke, M. and Warnecke, G. (2012) Exact Riemann solutions to compressible Euler equations in ducts with discontinuous cross-section. *J. Hyper. Differential Eq.* 9(3), 403-449
- Harten, A., Lax, P. D. and Leer, B. V. (1983) On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM review* 25(1), 35-61
- Henshaw, W. D., Smyth, N. F. and Schwendeman, D. W. (1986) Numerical shock propagation using geometrical shock dynamics. *Journal of Fluid Mechanics*, 171, 519-545
- Karlos, V., Solomos, G. and Larcher, M. (2016) Analysis of the blast wave decay coefficient using the Kingery-Bulmash data. *International Journal of Protective Structures*, 7(3), 409-429
- Kingery, C. N. (1966) Air blast parameters versus distance for hemispherical TNT surface bursts. Army Ballistic Research Lab, Aberdeen Proving Ground, USA

- Kröner, D. and Thanh, M. D. (2005) Numerical solutions to compressible flows in a nozzle with variable cross-section. *SIAM journal on numerical analysis*, 43(2), 796-824
- Lapebie E., Soulié, R. and Youinou, L. (2016) FLASH: Fast Lethality Assessment for Structures and Humans. 24<sup>th</sup> Int. Symp. Military Aspects of Blast and Shocks, Halifax, Canada.
- LeFloch, P. and Thanh, M. D. (2003) The Riemann problem for fluid flows in a nozzle with discontinuous cross-section. *Comm. Math. Sci.* 1(4), 763-797
- Ridoux, J., Lardjane, N., Monasse, L. and Coulouvrat, F. (2018) Comparison of geometrical shock dynamics and kinematic models for shock-wave propagation. *Shock Waves*, 28(2), 401-416
- Ruscade, G. (2021) Etude de la propagation des ondes de choc dans une double chambre. PhD Thesis of INSA Val de Loire, France
- Schwendeman, D. W. (1993) A new numerical method for shock wave propagation based on geometrical shock dynamics. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 441(1912), 331-341
- Thanh, M. D. (2009) The Riemann problem for a nonisentropic fluid in a nozzle with discontinuous cross-sectional area. *SIAM Journal on Applied Mathematics*, 69(6), 1501-1519
- Toro, E. F., Spruce, M. and Speares, W. (1994) Restoration of the contact surface in the HLL-Riemann solver. *Shock waves*, 4(1), 25-34
- Toro, E.F. (1997) *Riemann solvers and numerical methods for fluid dynamics: A practical introduction*. Springer, 1997
- Whitham, G. B. (2011) *Linear and nonlinear waves* (Vol. 42). John Wiley & Sons