



HAL
open science

Efficient 3D numerical prediction of the pressure wave generated by high-speed trains entering tunnels

David Uystepruyst, Mame William-Louis, Emmanuel Creusé, Serge Nicaise, François Monnoyer de Galland

► **To cite this version:**

David Uystepruyst, Mame William-Louis, Emmanuel Creusé, Serge Nicaise, François Monnoyer de Galland. Efficient 3D numerical prediction of the pressure wave generated by high-speed trains entering tunnels. *Computers and Fluids*, 2011, 47 (1), pp.165-177. 10.1016/j.compfluid.2011.03.005. hal-00768640

HAL Id: hal-00768640

<https://hal.science/hal-00768640>

Submitted on 22 Dec 2012

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.

Efficient 3D numerical prediction of the pressure wave generated by high-speed trains entering tunnels

David Uystepuyst^{a,b,*}, Mame William-Louis^{a,b}, Emmanuel Creusé^{a,c}, Serge Nicaise^{a,d}, François Monnoyer^{a,b}

^a*Université Lille Nord-de-France, F-59000 Lille, France.*

^b*TEMPO, Université de Valenciennes et du Hainaut-Cambrésis, 59313 Valenciennes Cedex 9, France.*

^c*Laboratoire Paul Painlevé, UMR CNRS 8524, and INRIA Lille Nord Europe (EPI SIMPAF), Université des Sciences et Technologies de Lille, Cité Scientifique, 59655 Villeneuve d'Ascq Cedex, France.*

^d*LAMAV, FR CNRS 2956, Université de Valenciennes et du Hainaut-Cambrésis, 59313 Valenciennes Cedex 9, France.*

Abstract

The topic of this paper is to present a new methodology for the three-dimensional numerical simulation of the entrance of high-speed trains in a tunnel. The movement of the train is made thanks to a technique of sliding meshes and a conservative treatment of the faces between two domains. All parts of the development are thought with the aim to reduce the computational time. In particular, non reflecting boundary conditions for non-structured three-dimensional meshes are developed in order to limit the calculation domain. Validations of the methodology are presented on different test cases.

Keywords:

*Corresponding author at: TEMPO, Université de Valenciennes et du Hainaut-Cambrésis, 59313 Valenciennes Cedex 9, France. Tel: +33 327511960 ; Fax: +33 327511961.

Email address: david.uystepuyst@univ-valenciennes.fr (David Uystepuyst)

Computational fluid dynamics, Euler equations, Three-dimensional simulation, Cartesian mesh, Sliding grid, Non-reflecting boundary conditions, High-speed trains.

1. Introduction

During the train entrance in a tunnel, the air initially inside the tunnel is strongly compressed by a piston effect in the train head neighborhood and gives rise to a compression wave that propagates at the sound speed in the tunnel. During its propagation in the tunnel, this wave can be damped in the case when the railway is on ballasted track, Ozawa [1], or can be steepened in the case when the railway is on concrete slab track, due to the nonlinear effects [2]. On reaching the distant tunnel exit, the compression wave is partially reflected back into the tunnel as a rarefaction wave, and a part of it emerges as a pressure pulse, called micro-wave. In certain circumstances this micro-wave may be strong enough to generate annoying sonic disturbances. Indeed, it can generate a booming noise up to 140-150 db or more, Maeda [3]. The magnitude and the duration of this wave are strongly connected with the temporal gradient of the initial compression wave. Hence it seems necessary to simulate with accuracy the generation of the compression wave both in amplitude and in temporal gradient. It is a challenging task, even for the current most advanced computers systems.

The flow generated by a train which circulates in a tunnel is three-dimensional, non stationary, turbulent and compressible. A complete study has to take into account all these aspects. However, comparing the results

obtained by Ogawa and Fujii [4] or by Shin and Park [5] that used the Navier-Stokes equations with the works of Réty and Grégoire [6, 7], Mok and Yoo [8] as well as Yoon [9], that have chosen an Eulerian approach, one realizes that the initial wave is perfectly simulated by both approaches. Actually, the pressure jump caused by the nose train entry does not depend on the viscosity [10], contrary to the second jump for which the wall friction effects are involved. However, this second jump, quasi-linear, does not modify the maximum pressure gradient. That is why Euler equations can be used to model this phenomenon.

During the displacement of the train, a relative movement with respect to the tunnel and to the ground can be distinguished. For the management of this movement, several techniques can be used:

- The chimera method, used by Ogawa [4] or Yoon [9] in a simplified version, also called overlapping meshes method. The computations are independently made in several subdomains and an interpolation has to be done for the transfer of the variables between the domains. To limit the interpolation errors, it is necessary to refine the common zones and consequently the computational cost can become very heavy, especially in the case of explicit time-integration schemes for which stability criteria induce small time steps.
- The method of meshes deformation, used by Réty [6, 7] or Mok [8] consists in building a new and more regular mesh during the simulation when the elements of the mesh become too stretched due to the deformations. An interpolation is also necessary to make the link between the old mesh and the new one.

- Finally, the technique of sliding meshes consists, as for the chimera method, to subdivide the computational domain into several subdomains, but without any intersection. This method is well suitable for a rectilinear movement like the one studied in this paper. Shin and Park [5] use this technique with an interpolation for the communication between domains.

During the simulations, the flow around the train is supposed to be stationary before it enters the tunnel. To accelerate the convergence towards this stationary state, non reflecting boundary conditions are needed to allow the reduction of the computational domain and to eliminate numerical disturbances.

This paper presents the development of a three-dimensional simulation based on the resolution of the Euler equations. The first originality of this work is to consider the displacement of the train by using a sliding meshes technique with a conservative treatment of the faces between two domains. The second one is to extend the non reflecting boundary conditions of Giles [11] to three-dimensional and unstructured meshes in this sliding meshes framework for general artificial boundaries. These contributions allow to get more accurate as well as low expensive numerical simulations, by reducing the computational domain without spurious non-physical reflection.

The paper is organized as follows: section 2 describes the methodology of sliding meshes. Then, section 3 is devoted to the model, the numerical scheme used, the non-reflecting boundary conditions implementation and their validation on an overpressure/depression test case. Finally, some validations

are performed on real cases to evaluate the jump of the pressure and of its temporal gradient induced by the entrance of a train in a tunnel. Here, experimental results [3, 6] confirm the reliability of the numerical methodology used in the present study.

2. Methodology of sliding meshes

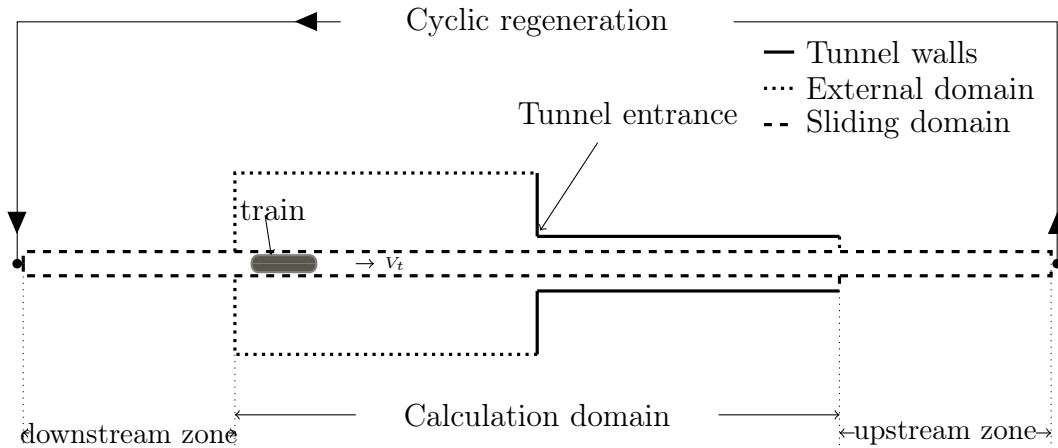


Figure 1: Top view of the computational domain.

The methodology [12] adopted here consists in subdividing the domain into two (non-overlapping) sub-domains. The first one (sliding domain part) contains the train and can be set in motion, as shown on Figure 1. The second domain encapsules the first one. It contains walls of the tunnel and the external domain. The sliding domain contains two additional parts: the "upstream zone" and the "downstream zone" whose limits are fixed. As the elements of the mesh arrive at the end of the "upstream zone", due to the movement of the domain, they are automatically re-injected at the beginning of the "downstream zone". With this approach, there is no need to regenerate

the mesh and the length of the sliding domain can be kept constant. Both sub-domains are independently meshed by means of an automatic Cartesian mesh generator [13] based on a boundary triangular mesh. The grid generator relies directly on an octree structure. The boundary cells are truncated and a merging procedure allows to avoid the convergence problem caused by small cells (see [13] for all details). Both sub-domains are connected to each other at common interfaces where a reconstruction of the cells faces is made on each side to preserve the conservative properties of the numerical scheme.

3. Equations and numerical scheme

3.1. Equations

As mentioned in the introduction, the pressure wave jump generated by the train entry in the tunnel is modeled by the 3D Euler equations, corresponding to the conservation of mass, quantity of movement and energy :

$$\partial_t \mathbf{U} + \nabla \cdot \mathbf{H}(\mathbf{U}) = 0, \quad (1)$$

where \mathbf{U} is the vector of conservative variables and $\mathbf{H}(\mathbf{U})$ the flux tensor:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix},$$

$$\mathbf{H}(\mathbf{U}) = \begin{pmatrix} \rho u & \rho v & \rho w \\ \rho u^2 + p & \rho uv & \rho uw \\ \rho vu & \rho v^2 + p & \rho vw \\ \rho wu & \rho wv & \rho w^2 + p \\ \rho uh_0 & \rho vh_0 + pv & \rho wh_0 \end{pmatrix}.$$

Here ρ stands for the volumic mass, u, v, w are the three components of the speed vector $\mathbf{V} = (u, v, w)^t$ and E is the energy per unit of mass.

Under the perfect gas assumption, the static pressure p is as usual given by:

$$p = (\gamma - 1)\rho \left[E - \frac{1}{2}(u^2 + v^2 + w^2) \right],$$

where $\gamma = 1.4$ and h_0 is the total enthalpy defined by:

$$h_0 = E + \frac{p}{\rho}.$$

Now, let us introduce the translation velocity of the sliding mesh $\mathbf{V}_t = (u_t, v_t, w_t)^t$, and the relative velocity of the fluid $\mathbf{V}_r = (u_r, v_r, w_r)^t = \mathbf{V} - \mathbf{V}_t$. In the sliding domain, equation (1) becomes:

$$\partial_t \mathbf{U} + \nabla \cdot \mathbf{H}^{(r)}(\mathbf{U}) = 0. \quad (2)$$

Where the fluxes tensor $\mathbf{H}^{(r)}(\mathbf{U})$ reads:

$$\mathbf{H}^{(r)}(\mathbf{U}) = \begin{pmatrix} \rho u_r & \rho v_r & \rho w_r \\ \rho u u_r + p & \rho u v_r & \rho u w_r \\ \rho v u_r & \rho v v_r + p & \rho v w_r \\ \rho w u_r & \rho w v_r & \rho w w_r + p \\ \rho u_r h_0 + p u_t & \rho v_r h_0 + p v_t & \rho w_r h_0 + p w_t \end{pmatrix}.$$

3.2. Numerical scheme

Integrating equation (1) on a control volume \mathcal{V} , we get

$$\partial_t \iiint_{\mathcal{V}} \mathbf{U} \, d\mathcal{V} + \iint_S \mathbf{H}_{\mathbf{n}}(\mathbf{U}) \, dS = 0,$$

where the volumic integral of the divergence of \mathbf{H} becomes a boundary integral due to Green-Ostrogradski formula, where $\mathbf{H}_{\mathbf{n}}(\mathbf{U}) = \mathbf{H}(\mathbf{U}) \cdot \mathbf{n}$, \mathbf{n} being the normal vector of the control volume.

After discretization of the computational domain, Roe's scheme [14] is used to determine the numerical approximation of the flux $\mathbf{H}_{\mathbf{n}}(\mathbf{U})$ at the interfaces of the control volumes. This numerical flux can be defined as the sum of a centered contribution and of a correction which depends on the eigenvectors and the eigenvalues of the Jacobian matrix $\partial \mathbf{H}_{\mathbf{n}}(\mathbf{U}) / \partial \mathbf{U}$. These last ones are well known, see Hirsch [15], in the case of the classical Euler equations. In order to calculate the numerical flux in the sliding domain, we have to re-determine them in the case of the Euler equations with a relative velocity. Let us denote $\mathbf{D}_{\mathbf{n}}$ the Jacobian matrix $\partial \mathbf{H}_{\mathbf{n}}(\mathbf{U}) / \partial \mathbf{U}$ and $\mathbf{D}_{\mathbf{n}}^{(r)}$ the Jacobian matrix $\partial \mathbf{H}_{\mathbf{n}}^{(r)}(\mathbf{U}) / \partial \mathbf{U}$. Clearly:

$$\mathbf{H}_{\mathbf{n}}^{(r)}(\mathbf{U}) = \mathbf{H}_{\mathbf{n}}(\mathbf{U}) - (V_t)_n \mathbf{U} ,$$

so that:

$$\mathbf{D}_{\mathbf{n}}^{(r)} = \mathbf{D}_{\mathbf{n}} - (V_t)_n \mathbf{I} ,$$

where \mathbf{I} is the identity matrix and $(V_t)_n = \mathbf{V}_t \cdot \mathbf{n}$.

Consequently, the eigenvalues, $\lambda_r^{(j)}$ ($1 \leq j \leq 5$), of $\mathbf{D}_{\mathbf{n}}^{(r)}$ are obtained by subtracting $(V_t)_n$ to the eigenvalues of $\mathbf{D}_{\mathbf{n}}$:

$$\lambda_r^{(1)} = \lambda_r^{(2)} = \lambda_r^{(3)} = (V_r)_n, \quad \lambda_r^{(4)} = (V_r)_n + c \quad \text{and} \quad \lambda_r^{(5)} = (V_r)_n - c, \quad (3)$$

where $c = \sqrt{\gamma p/\rho}$ is the sound speed.

The eigenvectors, useful for the construction of Roe’s scheme, remain the same.

With this formulation that takes into account the relative velocity, we can apply second order Roe’s scheme for the spatial discretization. The temporal integration is realized with a predictor-corrector scheme of Van Leer [16]. To preserve the TVD property of the scheme during the passage at the second order, the limiter for non-structured mesh of type Barth and Jespersen [17] is used.

3.3. Boundary conditions

3.3.1. Flux at the interface of the two meshes

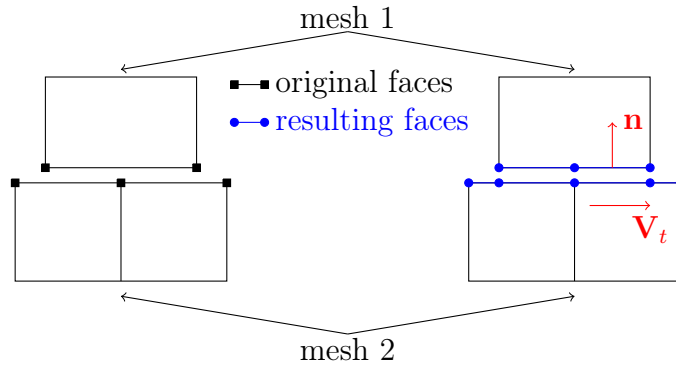


Figure 2: Construction of common interfaces.

In order to determine the fluxes between the two meshes, a calculation of the common interface (Figure 2), is made first, using the Sutherland-Hodgman [18] algorithm for convex polygon clipping. Figure 3 depicted an example of two polygons clipping. Starting with one edge of the first one, the separating line passing by its vertices is defined to subdivide the plane

of the polygon into the inside half-plane containing the first polygon and the remaining outside half plane. The result polygon becomes the intersection between the second polygon and the inside half-plane. This process is repeated for all edges of the first polygon.

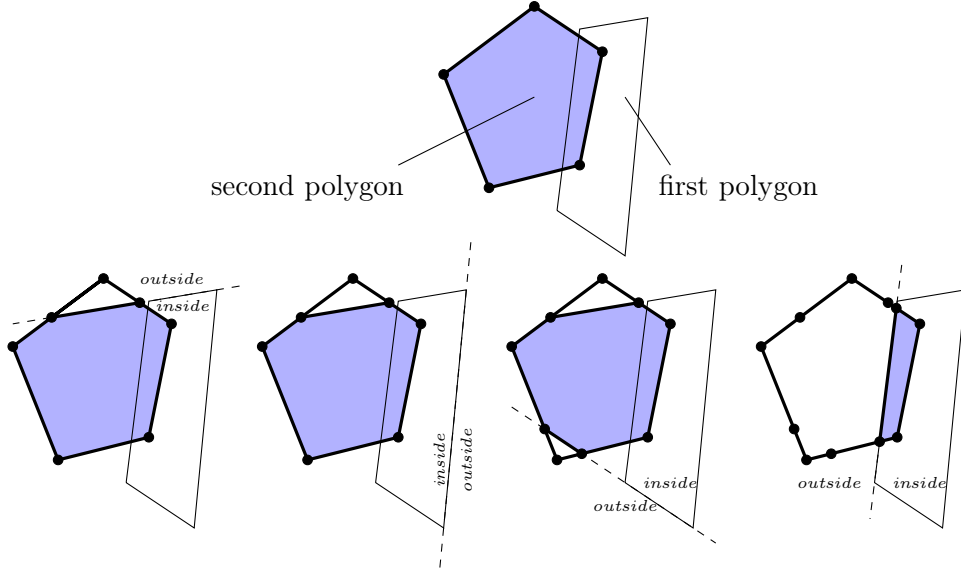


Figure 3: Common interface construction.

The translation velocity of the sliding mesh can a priori be considered as an additional difficulty. Nevertheless, by definition of the sliding meshes technique, the normal vectors of the interfaces are always orthogonal to the translation vector of the mesh, and thus

$$\mathbf{V}_t \cdot \mathbf{n} = 0.$$

So the translation velocity does not interfere in the calculation of the flux at the interfaces. Hence a classical flux computation can be performed there. This allows a conservative calculation of the fluxes at the interfaces between

the two meshes.

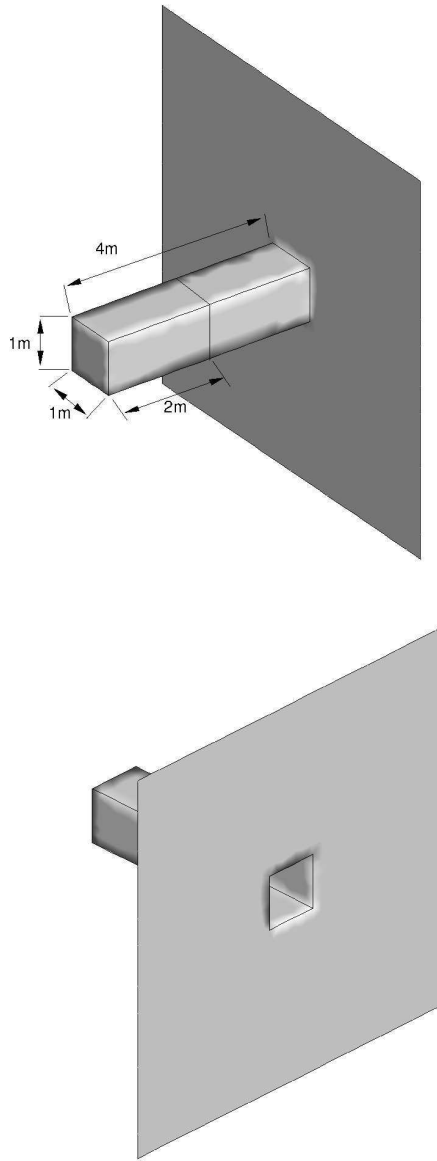


Figure 4: Shock tube definition.

The validation of the method is performed on the shock tube defined on

Figure 4. It is a 4 m length tube separated into two parts by a diaphragm in its middle, with the closed side initially set at a pressure of 1.25 bar. The other part is opened to the free atmosphere.

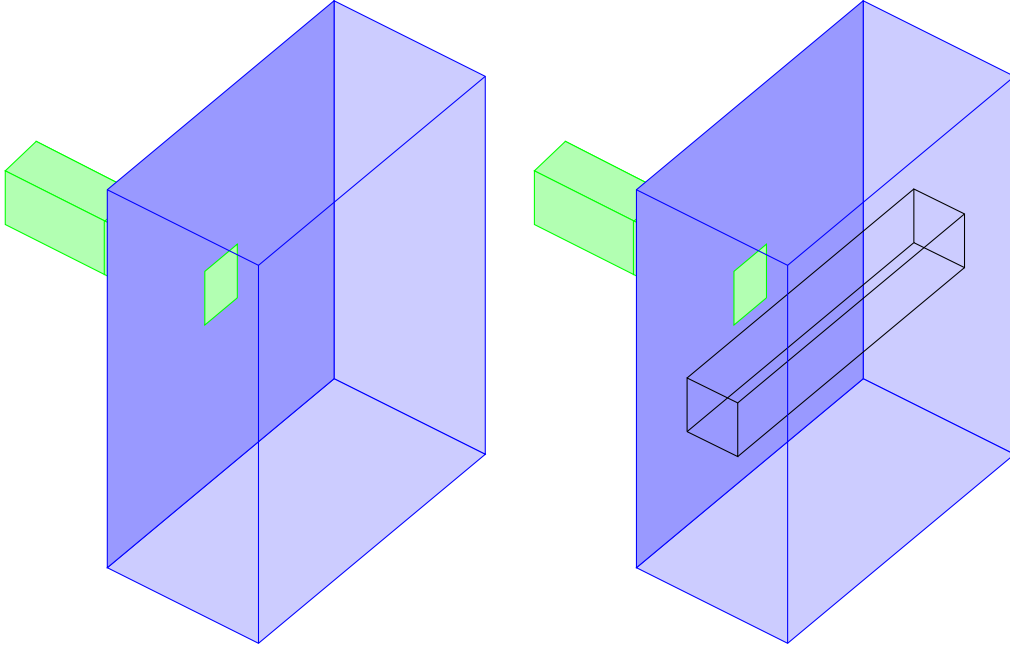


Figure 5: Single block model (left) and configuration with sliding block (right).

Two configurations have been considered, as shown on Figure 5. The first one consists in a single domain, while the second one is subdivided into two domains. In this configuration, the second domain can be set in motion.

In order to measure the capability of the method to handle the propagation of strong gradients across the sliding interface, a numerical sensor is located in the fixed domain just behind the sliding domain, represented by a cross into Cartesian meshes on Figure 6. On this figure, the sliding domain is slightly longer than the fixed domain to ensure the cyclic regeneration of

cells.

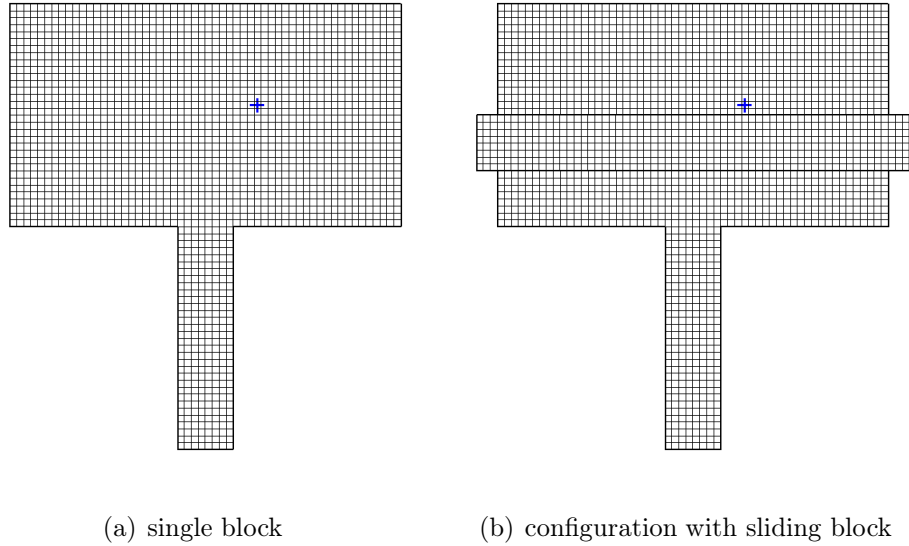


Figure 6: Sketch of shock tubes Cartesian grids.

The calculation on the second configuration is performed at translation speed of 0 and 70 m/s. Results of the non-dimensional pressure $p_* = (p - p_\infty) / p_\infty$ (with $p_\infty = 1 \text{ atm}$), recorded at the numerical sensor, are given on Figure 7.

Qualitatively, it can be shown that the conservative method is in very good accordance with the reference solution obtained with the single block calculation, both v equal 0 and v equal 70 m/s. In order to quantify these results the L^∞ norm of non-dimensional pressure errors are summarized in Table 1.

When the sliding domain does not move, the obtained solution has to be the same than the solution of the single block. The existing small error is due to the fact that volumic meshes are slightly different between both

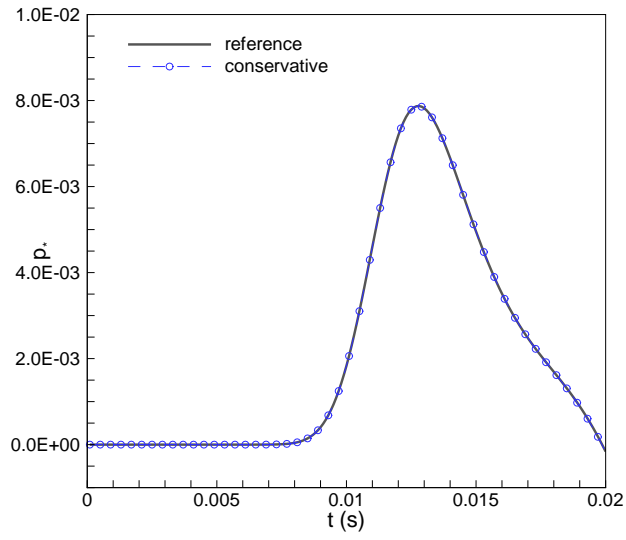
	$\ p_* - (p_*)_{ref}\ _\infty$	$\ p_* - (p_*)_{ref}\ _\infty / \ (p_*)_{ref}\ _\infty$
v=0 m/s	9.87×10^{-6}	1.29×10^{-3}
v=70 m/s	7.83×10^{-5}	1.03×10^{-2}

Table 1: Infinite norms of the non-dimensional pressure errors, sliding mesh validation.

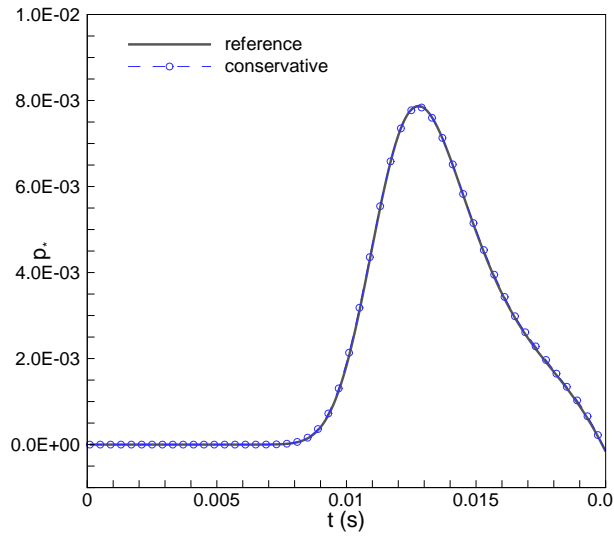
configurations. At the speed of 70 m/s, the error slightly increases. Since the timestep used for the calculations is the same for both configurations, the CFL number differs due to the translation velocity of the sliding block. However, the relative error remains only about 1%.

Now, let us consider a convergence analysis. Without any available analytical solution, a calculation is performed on the single block with a very fine mesh size $\Delta x = 1/32 m$ which leads to 6 553 600 cells. Then, seven calculations are made with space mesh size $\Delta x = 1/2(i+1) m$ for $i = 1, \dots, 7$ for the single block and for the domain with sliding block, with the translation velocity equal to 70 m/s. The evolution of the L^∞ norm of the non-dimensional pressure error is depicted on the Figure 8 versus the space mesh size.

It can be showed that the sliding grid method gives the same convergence order than the single block. Moreover, the sliding domain curve gets closer to the single block curve while the space mesh size decreases.



(a) $v=0$ m/s



(b) $v=70$ m/s

Figure 7: Temporal variation of non-dimensional pressure, sliding mesh validation.

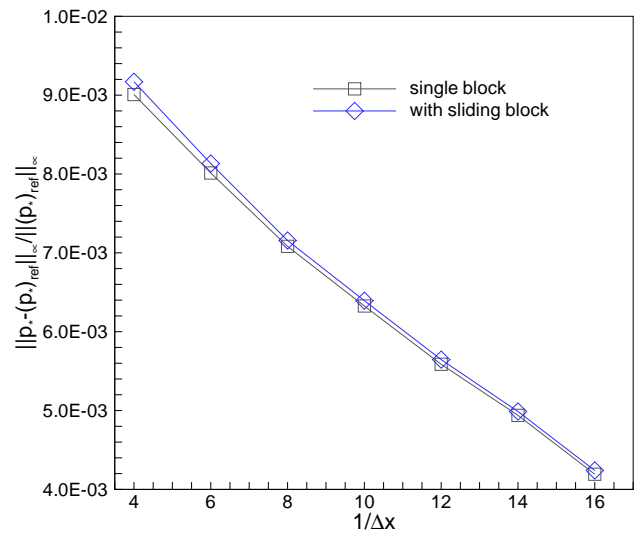


Figure 8: Convergence analysis of the sliding mesh validation.

3.3.2. Non reflecting boundary conditions

The non reflecting boundary conditions allow to make the calculations without the appearance of non physical reflections. Giles [11] introduced such a type of boundary conditions for the two-dimensional Euler equations and in the case of boundary faces parallel to one axis of the domain. These conditions are based on a Fourier decomposition of the solution at the boundary. Then Nicoud extended Giles' conditions for arbitrary boundary faces in the case of a subsonic 2D [19] and 3D [20] outflow.

The boundary condition presented here is defined for an arbitrary three-dimensional face and applied to outflow as well as inflow boundaries. Furthermore, the translation velocity of the sliding domain is taken into account. Practically, let us suppose that we have to define the fluxes through a face centered in F (see Figure 9) belonging to a boundary on which some non reflecting boundary conditions are imposed (see for instance the boundary of the external domain in dotted line on Figure 1). For that purpose, Euler equations are written in a characteristic formulation. First, let us define the three fluxes $\mathbf{F}^{(r)}(\mathbf{U})$, $\mathbf{G}^{(r)}(\mathbf{U})$ and $\mathbf{K}^{(r)}(\mathbf{U})$, respectively, as the first, the second and the third column of the flux tensor $\mathbf{H}^{(r)}(\mathbf{U})$. Equation (2) can be rewritten:

$$\partial_t \mathbf{U} + \mathbf{A}^{(r)} \partial_x \mathbf{U} + \mathbf{B}^{(r)} \partial_y \mathbf{U} + \mathbf{C}^{(r)} \partial_z \mathbf{U} = 0, \quad (4)$$

where $\mathbf{A}^{(r)}$, $\mathbf{B}^{(r)}$, $\mathbf{C}^{(r)}$ are the corresponding Jacobian matrices $\frac{\partial \mathbf{F}^{(r)}(\mathbf{U})}{\partial \mathbf{U}}$, $\frac{\partial \mathbf{G}^{(r)}(\mathbf{U})}{\partial \mathbf{U}}$, and $\frac{\partial \mathbf{K}^{(r)}(\mathbf{U})}{\partial \mathbf{U}}$. Let us denote by \mathbf{n} the normal vector to this face and let the matrix $\mathbf{D}_{\mathbf{n}}^{(r)} = \mathbf{A}^{(r)} n_x + \mathbf{B}^{(r)} n_y + \mathbf{C}^{(r)} n_z$. This matrix can be

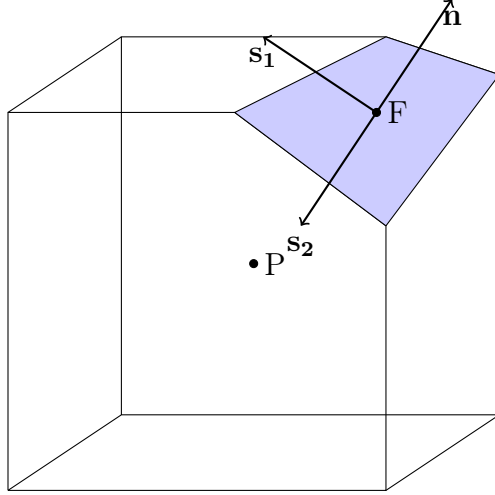


Figure 9: Boundary cell.

diagonalized, with the eigenvalues given by (3):

$$\mathbf{\Lambda}^{(r)} = \mathbf{L}\mathbf{D}_{\mathbf{n}}^{(r)}\mathbf{L}^{-1} = \text{diag}(\lambda_r^{(1)}, \lambda_r^{(2)}, \lambda_r^{(3)}, \lambda_r^{(4)}, \lambda_r^{(5)}).$$

Characteristic variables are defined by the relation valid for arbitrary variations δ , see Hirsch [15]:

$$\delta\mathbf{W} = \mathbf{L}\delta\mathbf{U}, \quad \delta\mathbf{U} = \mathbf{L}^{-1}\delta\mathbf{W}.$$

Premultiplying (4) by \mathbf{L} , we obtain the five equations:

$$\left\{ \begin{array}{l} \partial_t W_1 + \mathbf{V}_r \cdot \nabla W_1 = 0 \\ \partial_t W_2 + \mathbf{V}_r \cdot \nabla W_2 + \frac{1}{2}c\mathbf{s}_1 \cdot (\nabla W_4 + \nabla W_5) = 0 \\ \partial_t W_3 + \mathbf{V}_r \cdot \nabla W_3 + \frac{1}{2}c\mathbf{s}_2 \cdot (\nabla W_4 + \nabla W_5) = 0 \\ \partial_t W_4 + (\mathbf{V}_r + c\mathbf{n}) \cdot \nabla W_4 + c(\mathbf{s}_1 \cdot \nabla W_2 + \mathbf{s}_2 \cdot \nabla W_3) = 0 \\ \partial_t W_5 + (\mathbf{V}_r - c\mathbf{n}) \cdot \nabla W_5 + c(\mathbf{s}_1 \cdot \nabla W_2 + \mathbf{s}_2 \cdot \nabla W_3) = 0, \end{array} \right.$$

where $(\mathbf{n}, \mathbf{s}_1, \mathbf{s}_2)$ forms an orthonormal basis, see Figure 9. In a full expansion, it is equivalent to

$$\left\{ \begin{array}{l} \partial_t W_1 + (V_r)_n \partial_n W_1 + (V_r)_{s_1} \partial_{s_1} W_1 + (V_r)_{s_2} \partial_{s_2} W_1 = 0 \\ \partial_t W_2 + (V_r)_n \partial_n W_2 + (V_r)_{s_1} \partial_{s_1} W_2 + (V_r)_{s_2} \partial_{s_2} W_2 + \frac{1}{2}c(\partial_{s_1} W_4 + \partial_{s_1} W_5) = 0 \\ \partial_t W_3 + (V_r)_n \partial_n W_3 + (V_r)_{s_1} \partial_{s_1} W_3 + (V_r)_{s_2} \partial_{s_2} W_3 + \frac{1}{2}c(\partial_{s_2} W_4 + \partial_{s_2} W_5) = 0 \\ \partial_t W_4 + ((V_r)_n + c)\partial_n W_4 + (V_r)_{s_1} \partial_{s_1} W_4 + (V_r)_{s_2} \partial_{s_2} W_4 + c(\partial_{s_1} W_2 + \partial_{s_2} W_3) = 0 \\ \partial_t W_5 + ((V_r)_n - c)\partial_n W_5 + (V_r)_{s_1} \partial_{s_1} W_5 + (V_r)_{s_2} \partial_{s_2} W_5 + c(\partial_{s_1} W_2 + \partial_{s_2} W_3) = 0. \end{array} \right. \quad (5)$$

The outflow condition of Giles, extended by Nicoud [20] for a 3D arbitrary face is:

$$\partial_t W_5 = -(V_r)_{s_1} \partial_{s_1} W_5 - (V_r)_{s_2} \partial_{s_2} W_5 - (V_r)_n (\partial_{s_1} W_2 + \partial_{s_2} W_3). \quad (6)$$

Using the fifth equation of (5), condition (6) can be rewritten in its spatial form:

$$\partial_n W_5 = \partial_{s_1} W_2 + \partial_{s_2} W_3. \quad (7)$$

With the strategy used by Nicoud, we extend the original Giles' conditions to an arbitrary three dimensional boundary face. Results obtained are given in Table 2. Those conditions are now named NRBC (Non Reflecting Boundary Conditions). Those expressions allow us to make the extrapolation discussed below.

In order to define the fluxes at the boundary, \mathbf{U}_F is defined from \mathbf{U}_P using a suitable extrapolation (see Figure 9), that reads:

$$\begin{aligned} \mathbf{U}_F &= \mathbf{U}_P + (\nabla \mathbf{U})_P \cdot \mathbf{PF} \\ &= \mathbf{U}_P + (\partial_n \mathbf{U})_P \mathbf{PF} \cdot \mathbf{n} + (\partial_{s_1} \mathbf{U})_P \mathbf{PF} \cdot \mathbf{s}_1 + (\partial_{s_2} \mathbf{U})_P \mathbf{PF} \cdot \mathbf{s}_2. \end{aligned}$$

Flow	Conditions
outgoing	$\partial_n W_5 = \partial_{s_1} W_2 + \partial_{s_2} W_3$
incoming	$\partial_n W_1 = 0$ $\partial_n W_2 = \partial_{s_1} W_2$ $\partial_n W_3 = \partial_{s_2} W_3$ $\partial_n W_4 = -\frac{1}{2}(\partial_{s_1} W_2 + \partial_{s_2} W_3)$

Table 2: Spatial formulations of NRBC conditions.

$(\partial_{s_1} \mathbf{U})_P$ and $(\partial_{s_2} \mathbf{U})_P$ are evaluated by a discretization of the gradient inside the domain, $(\partial_n \mathbf{U})_P$ is evaluated by the knowledge of the spatial variation of the characteristic variables from $(\partial_n \mathbf{U})_P = \mathbf{L}^{-1}(\partial_n \mathbf{W})_P$.

Two cases have to be considered:

- *Outflow*: Only the fifth characteristic variable leaves the domain: $\partial_n W_j$ ($1 \leq j \leq 4$) are evaluated by a discretization inside the computational domain and $\partial_n W_5$ is derived from $\partial_{s_1} W_2 + \partial_{s_2} W_3$ using the Giles conditions (Table 2).
- *Inflow*: We proceed similarly using this Table 2 to express $\partial_n W_j$ ($1 \leq j \leq 4$) and $\partial_n W_5$ is discretized inside the domain.

To validate these non reflecting boundary conditions, a domain D_0 defined as the unit cube is considered. This domain is discretized by 100 elements in each direction. As initial condition, a slightly de-centered Gaussian function

of pressure is imposed:

$$p_0 = \begin{cases} p_\infty \left(1 + \mu e^{-1/(\sqrt{x^2+(y-0.15)^2+z^2}-0.3)^2} \right) & \text{for } \sqrt{x^2 + (y - 0.15)^2 + z^2} < 0.3, \\ p_\infty \text{ else,} & \\ u_0 = v_0 = w_0 = 0 \text{ m.s}^{-1}, & \\ T_0 = 293 \text{ K,} & \\ \rho_0 = \frac{p_0}{R T_0}, & \end{cases} \quad (8)$$

where $p_\infty = 101325 \text{ Pa}$.

Equation (1) is solved with this initial condition.

This initialization condition implies an overpressure followed by a depression. Hence arrived at the limit of the domain, the flow is at first outgoing and then ingoing. This allows to validate both conditions given in Table 2. In order to evaluate the results obtained on this domain D_0 , a reference solution was computed using a larger domain which can be considered as "infinite", since the waves do not reach its boudaries during the simulation time involved.

Graphs from Figure 10 show the propagation of the wave generated by the initial data (8). The reference value is represented by the black continuous lines, while the value given by the calculation onto the truncated domain is represented by dotted lines blue and circles. The isolines of non-dimensional pressure $p_* = (p - p_\infty) / p_\infty$ on the plane $x = 0$ (top) as well as the difference of pressure on the lines $x = z = 0$ (bottom right) and $x = z = 0,25 \text{ m}$ (bottom left) are shown at time $t = 0,0015 \text{ s}$. The Giles' formulation is

compared with a simple extrapolation method:

$$\left\{ \begin{array}{l} \rho_F = \rho_P + \left(\vec{\nabla} \rho \right)_P \cdot \vec{PF} \\ u_F = u_P + \left(\vec{\nabla} u \right)_P \cdot \vec{PF} \\ v_F = v_P + \left(\vec{\nabla} v \right)_P \cdot \vec{PF} \\ w_F = w_P + \left(\vec{\nabla} w \right)_P \cdot \vec{PF} \\ p_F = p_P + \left(\vec{\nabla} p \right)_P \cdot \vec{PF}, \end{array} \right.$$

which consists in defining the variables values at the face center F from the variables values at the cell center P and their gradients, all of them being evaluated by their values inside the computational domain. Results of this method are represented by green lines and triangles.

The two lines allow to show the efficiency of the boundary conditions of NRBC in the case of an almost one-dimensional exit, where the wave is almost parallel to the boundary: right $x = z = 0 \text{ m}$; and a multidimensional exit, where the wave arrives with a non zero angle of attack: $x = z = 0,25 \text{ m}$. It can be seen that the extrapolation method implies an important drift.

Table 3 shows the quantitative results. The error generated by the NRBC method is about four times lower than the error of the extrapolation method.

	$\ p_* - (p_*)_{ref}\ _\infty / \ (p_*)_{ref}\ _\infty$	
	NRBC method	Extrapolation method
$x = z = 0$	1.92×10^{-2}	7.86×10^{-2}
$x = z = 0.25$	5.71×10^{-2}	2.31×10^{-1}

Table 3: Relative error norms of non-dimensional pressure, NRBC validation.

Finally, as noticed on the various graphs and on the error on Table 3,

values obtained on the truncated domain with the NRBC method are in good accordance with the reference values and no significant reflection can be seen.

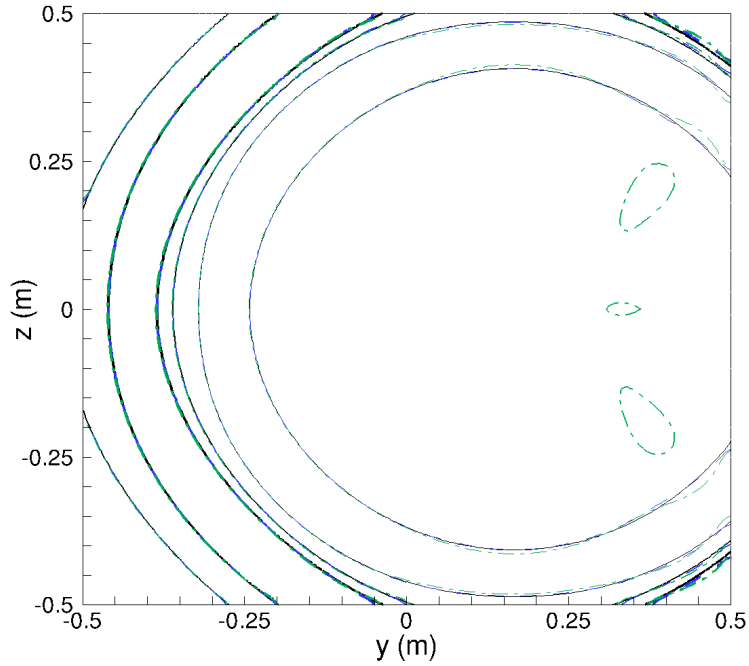
	domain D_0 extrapolation method	domain D_0 NRBC method	infinite domain
CPU time (s)	1152	1195	9433

Table 4: Calculations CPU time (in second).

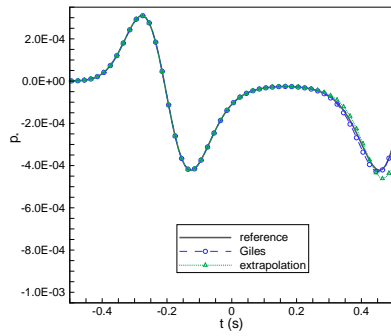
Table 4 indicates requisite CPU time for those simulations. The calculation on the domain D_0 , with Giles' method, is slightly more time consuming than the extrapolation method on this same domain. Nevertheless, the calculation on the "infinite" domain, which is two times larger than D_0 in each direction, requires about 8 times more computational efforts. With the degree of convenience between results obtained on domain D_0 with Giles' method and on the "infinite" domain, it is possible to consider domain D_0 rather than the "infinite" domain and, then, to strongly reduce the computational time.

In order to realize a convergence analysis, the reference solution obtained on the "infinite" domain with a mesh size of $\Delta x = 1/100 m$ is compared with numerical solutions resulting of calculations performed on the domain D_0 with mesh size $\Delta x = 1/(20 * i) m$ for $i = 1, \dots, 4$. Results are displayed on Figure 11. This graph gives the relative L^∞ norm error of the non-dimensional pressure time history for the sensor coordinates (0.25;0.5;0.25), located at the boundary. It can be shown the error decreases

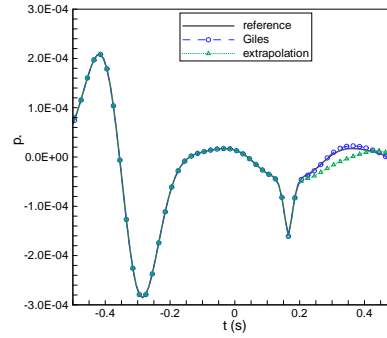
with the mesh size.



(a) $x = 0$



(b) $x = z = 0,25$



(c) $x = z = 0$

Figure 10: Comparison of NRBC method and extrapolation method with reference data. (a) Isocontours pressure: (—) numerical results on "infinite" domain, (---) numerical results of NRBC method on domain D_0 , (-.-) numerical results of extrapolation method on domain D_0 .

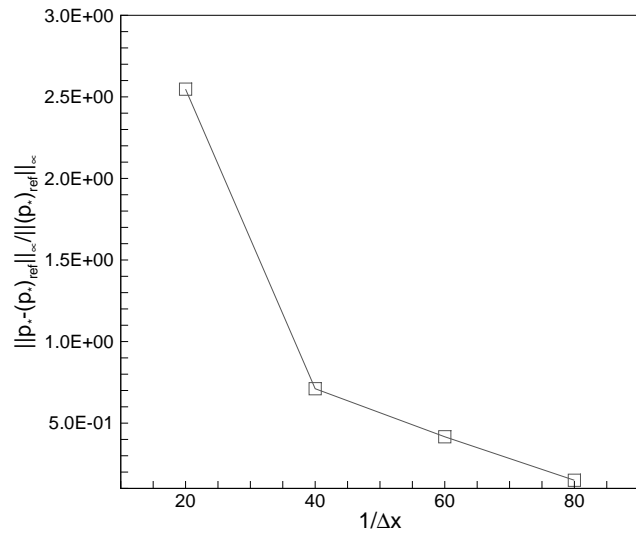


Figure 11: Evolution of the non-dimensional pressure error of a sensor located at $x=z=0.25$ and $y=0.5$, NRBC validation.

3.3.3. Solid walls

In order to consider the correction of the flux of the numerical scheme for faces situated on the boundary of the domain, a ghost cell is built in which the flow velocity \mathbf{V}_{ghost} is determined thanks to the flow speed in boundary cell \mathbf{V}_b :

$$(\mathbf{V}_{ghost})_t = (\mathbf{V}_b)_t \text{ and } (\mathbf{V}_{ghost})_n = -(\mathbf{V}_b)_n$$

The other variables are identical. In that way the impermeability condition is satisfied while introducing the flux correction of Roe.

4. High speed train validations

4.1. Reduced scale tests

To study the influence of the nose shape of the train on the initial compression wave, Maeda [3] made measurements on a reduced scale model. The body of the reduced train is cylindrical and three shapes of the nose are considered: conical, parabolic and elliptic.

As stated on Figure 12, these three forms imply a different variation of section. The tunnel is a cylindrical tube with one sensor of pressure located at 1 m from the entrance. The blocking ratio $\sigma = S_{tr}/S_{tun}$, defined as the maximum cross-sectional area of the train to the one of the tunnel, is 0.116 and the train is launched at 232 *km/h*, which corresponds for an ambient temperature of 20° C at a Mach number of 0.188. Due to the symmetry of the configuration, only the quarter of the domain has to be considered. So the number of cells is about 300 000. Figure 13 represents the triangular surfacic meshes of the three configurations, and Figure 14 represents the corresponding volumic cartesian meshes. The numerical results of the jump

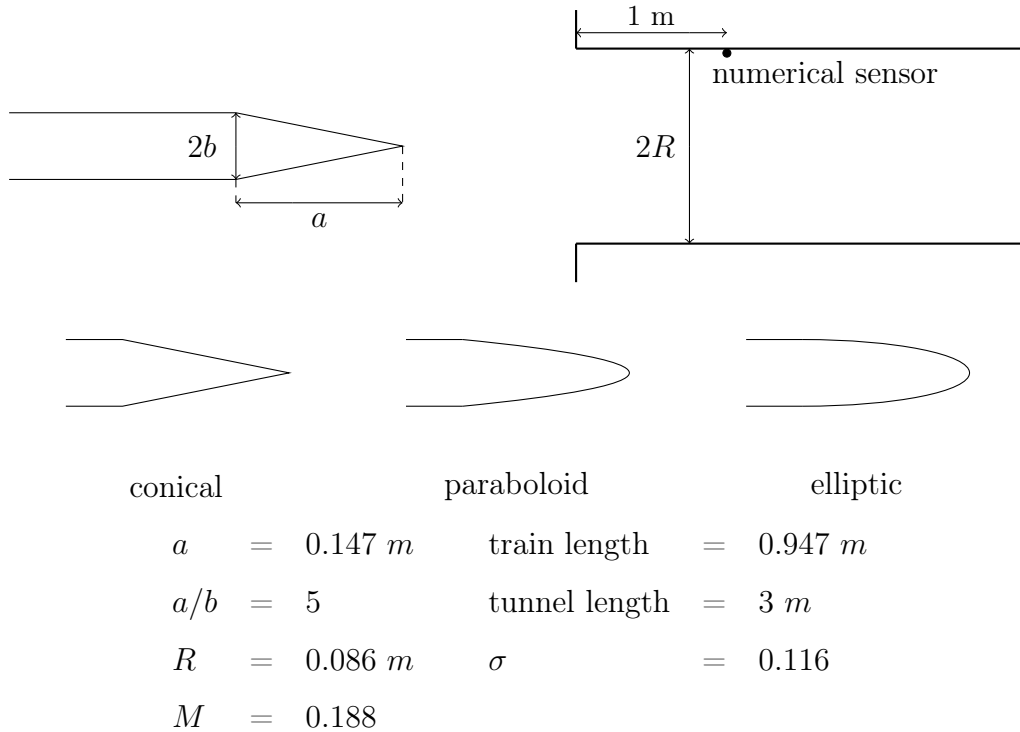


Figure 12: Geometrical ratios and form of the nose.

of pressure and of its temporal gradient recorded at the sensor are compared with the experimental results on the graphs of Figure 15.

Graphs as well as maxima presented in Table 5 show a very good concordance between the experimental and numerical results. The errors on the jumps of pressure do not exceed 6%, while the errors on the maxima of gradients of pressure are lower than 4%. We notice that the committed errors are more important for the conical profile than for two other cases. The geometrical discontinuity of the conical nose, between nose and body, implies this most important error.

Although the shape of the nose has little influence on the jump of pressure, it

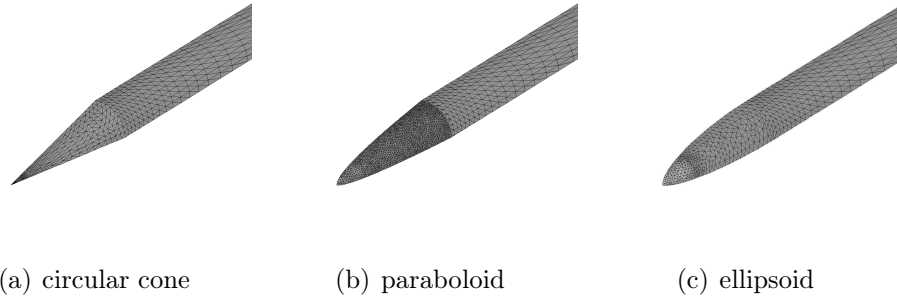


Figure 13: Surfacic meshes.

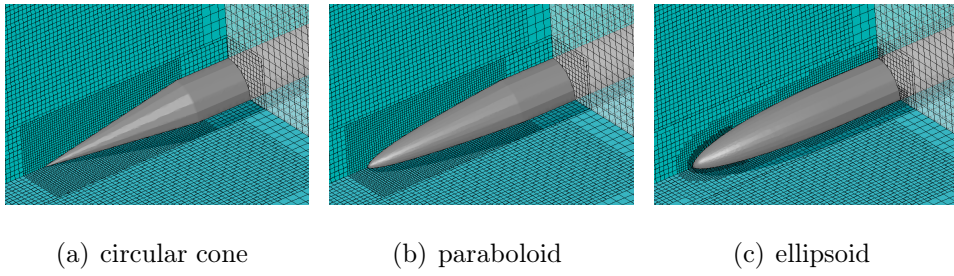


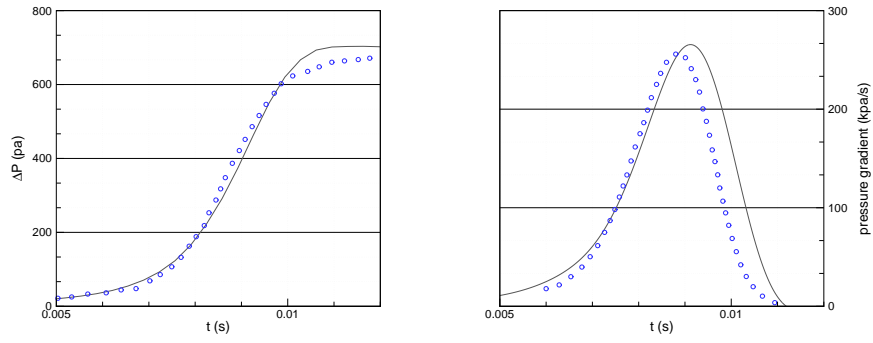
Figure 14: Volumic cartesian meshes.

shape		ΔP max		$\partial P/\partial t$ max	
		value (pa)	diff./ref. (%)	value (kpa/s)	diff./ref. (%)
cone	exp.	660		255	
	num.	702	+6	265	+4
paraboloid	exp.	684		232	
	num.	712	+4	229	-1
ellipsoid	exp.	689		262	
	num.	717	+4	259	-1

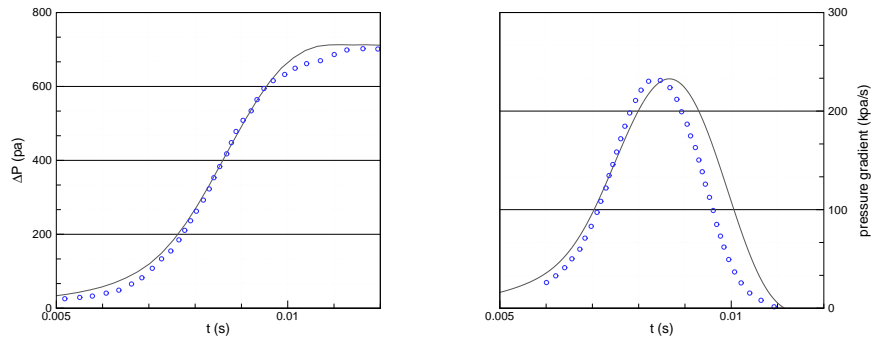
Table 5: Pressure and its gradient maxima and relative error.

is noticed that the paraboloid profile clearly minimizes the maximum of the pressure gradient (see Figure 16). This is due to the fact that the variation of the cross-sectional area of the parabolic nose is constant (see the variations of the areas of the sections of shape of the three configurations represented on Figure 17(b)).

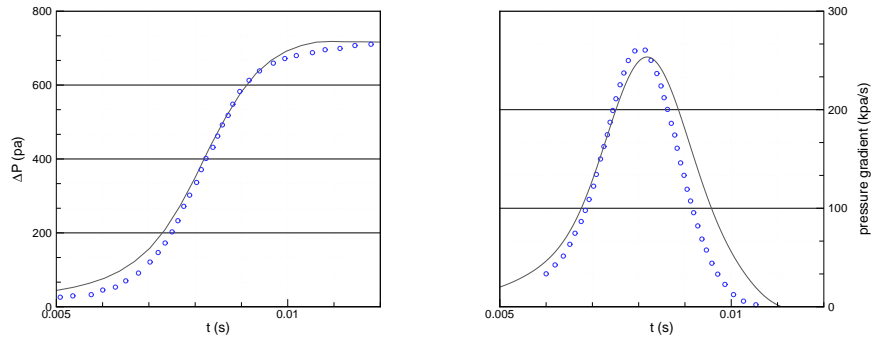
The compression wave is delayed, it appears by chronological order: ellipse, parabola, cone, as shown on Figure 16. For a given value of x on the nose, the section of shape of maximal area is the one of the elliptic nose, as shown on Figure 17. At this point, the pressure is thus more important for the ellipse. This value of pressure is reached later for the parabola, and even later for the cone. This explain the temporal shift of the ascent for the pressure gradient for the three configurations.



(a) circular cone

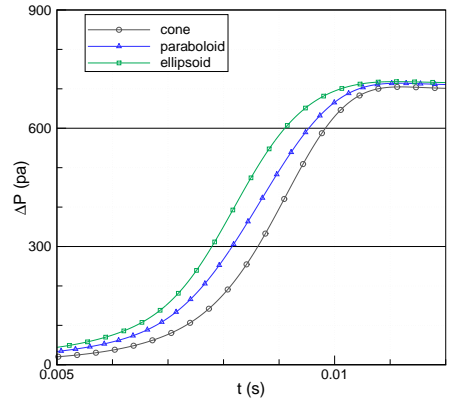


(b) paraboloid of revolution

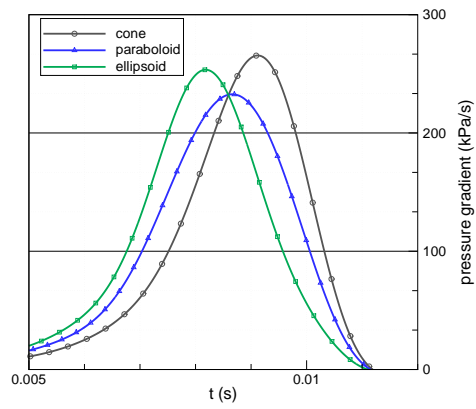


(c) ellipsoid of revolution

Figure 15: Comparison of jumps (left) and gradients (right) of pressure for the 3 shapes of the nose; $\circ \circ \circ$: experimental results, $—$: numerical results.

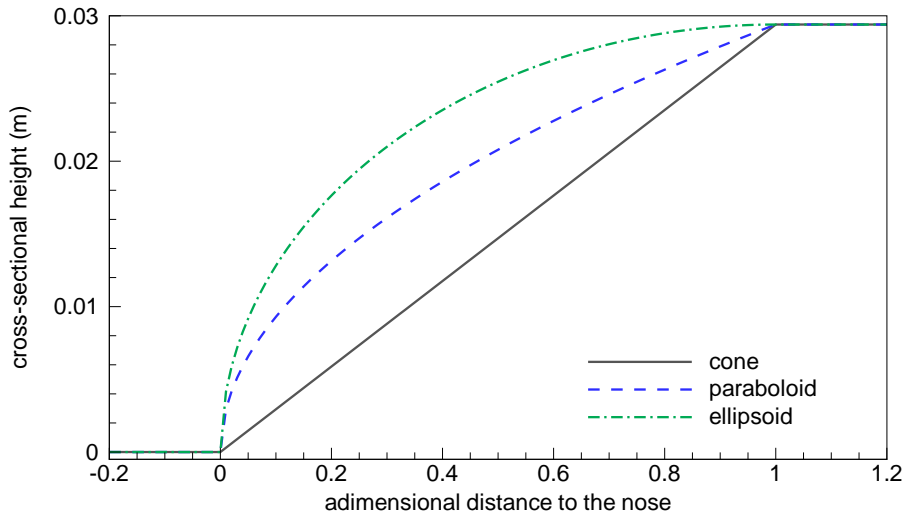


(a) pressure amplitude

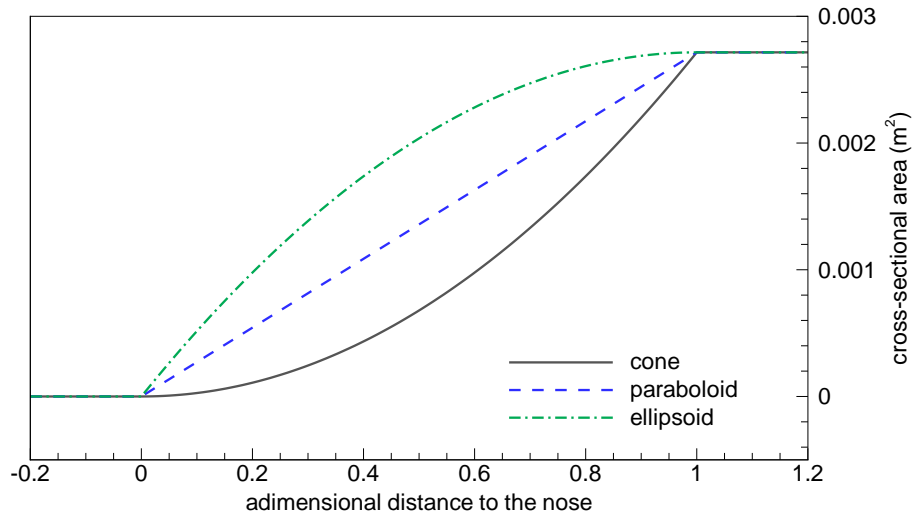


(b) pressure gradient

Figure 16: Comparison between the numerical results of the three nose shapes.



(a) nose shape



(b) cross-sectional area

Figure 17: Section variation.

4.2. Full scale test

The full scale test considered here is based on the experiment realized *in situ* by the SNCF (French National Railway Company) [6]. This experiment consists in the entrance of the French Hight Speed Train (TGV) into the tunnel of Villejuste situated on the high-speed southwest line, in France. The train measures 74 m and runs at a speed of 220 km.h⁻¹ before entering in a tunnel of section of area of 46 m², that gives a blocking ratio of $\sigma = S_{tr}/S_{tun} = 0.21$.

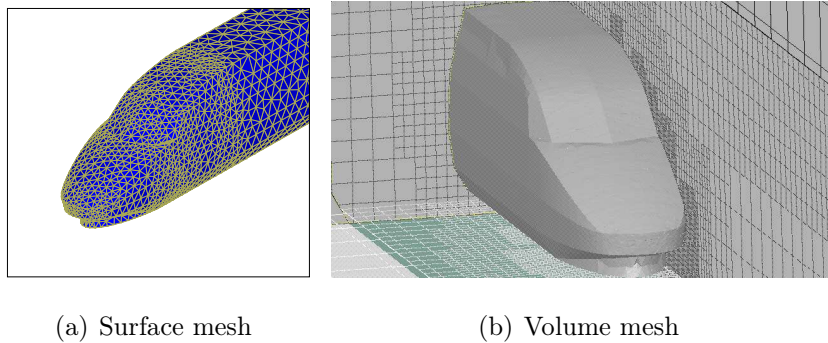


Figure 18: Volumic and surfacic meshes of the TGV.

The interest of this experiment is that our results can be compared with the experimental data but also with the numerical results obtained by the SNCF using the software Tg-flo (three-dimensional Eulerian code).

4.2.1. Tg-flo numerical setup

The mesh is composed of tetrahedra and is unstructured. An arbitrary Lagrange-Euler formulation, which allows to take into account the relative movement between the tunnel and the train, is used to solve the Euler equations using a second order Runge-Kutta scheme for the temporal discretization, and a finite element method for the spatial one.

A complete simulation is subdivided in two phases. The first one consists in the steady state calculation, with a motionless train. The second one corresponds to the train entry in the tunnel. At the beginning of the displacement, the train is located 15 m downstream of the tunnel.

In order to make this second phase, a mesh deformation is used and a remeshing is done when elements become too small or too stretched.

For the transient simulation (second phase), the computational domain is subdivided in two thanks of a symmetry condition, this leads to a cells number of about 200 000. The minimum space mesh size is about 0.1 m.

4.2.2. Metas numerical setup

Concerning Metas, no steady state is performed before the train displacement. Therefore, the train-tunnel distance have to be long enough to obtain this steady state before the train entry. In addition, if the train-tunnel distance is too short, the sudden train displacement implies a contamination of the pressure signal. The initial train-tunnel distance is 160 m, against 15 m for Tg-flo. Thus, the domain is twice as long as the Tg-flo domain. The overall physical domain is modeled and no symmetry is used.

For the first calculation, meshes contain about 500 000 elements, and the minimum space mesh size is 0.2 m. This mesh is globally more coarse than Tg-flo mesh.

The calculation Metas2 was performed to obtain a better result. To realize that, Metas1 mesh is refined at two important locations: train nose and tunnel entry. With this refinement, the cells number reaches 1 360 000 and the minimum space mesh size is as for Tg-flo equal to 0.1 m.

All the numerical data are summarized in the Table 6.

	Tg-flo (SNCF)	Metas1	Metas2
domain	half-domain	full domain	full domain
dist. train-tunnel at t=0	$\simeq 15$ m	160 m	160 m
nbr of cells	200 000	500 000	1 360 000
$(\Delta x)_{min}$	0.1 m	0.2 m	0.1 m

Table 6: Data relative to the numerical experiments, full scale test.

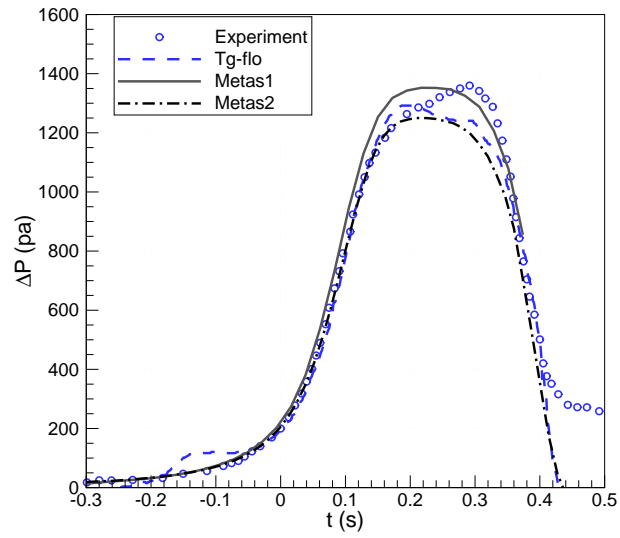
4.2.3. Results

The graphs of Figure 19 compare the different results for the jump of pressure and the temporal gradient of pressure, obtained by the experiment result, the numerical solution of the SNCF code and finally Metas simulations. The values of the maxima are presented in Table 7.

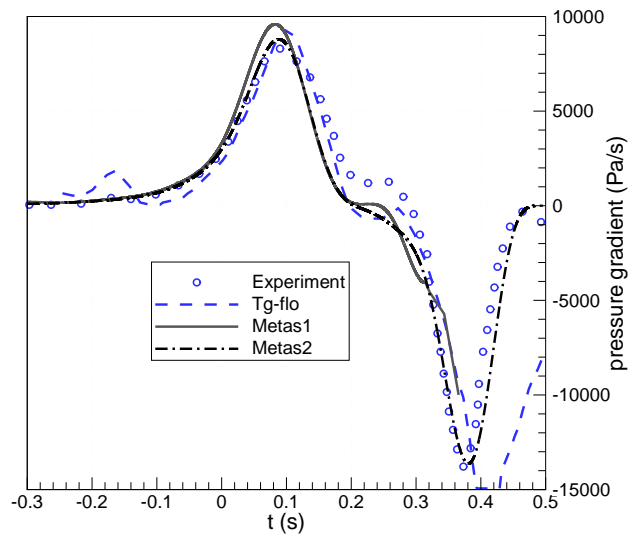
	ΔP max		$\partial P/\partial t$ max	
	value (pa)	diff./ref. (%)	value (pa/s)	diff./ref. (%)
experimental	1296		8400	
Tgflo	1291	-1	9330	+11
Metas1	1332	+3	9600	+14
Metas2	1251	-3	8795	+4

Table 7: Pressure and its gradient maxima and relative error.

The determination of the pressure amplitude is not easy from the experimental data. Indeed, as previously mentioned, the pressure signal is



(a) Pressure amplitude



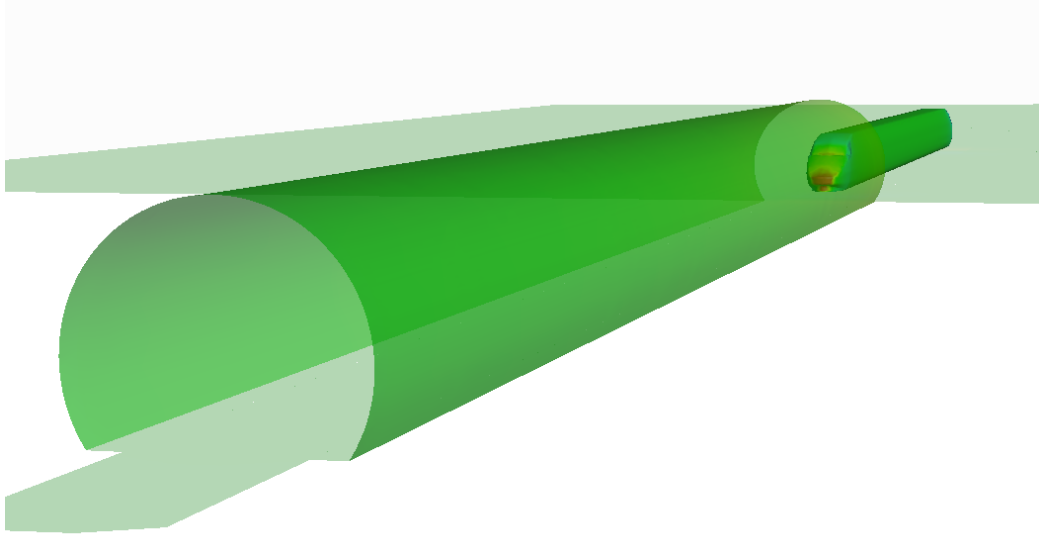
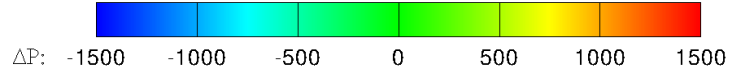
(b) Temporal pressure gradient

Figure 19: History of the pressure and temporal pressure gradients.

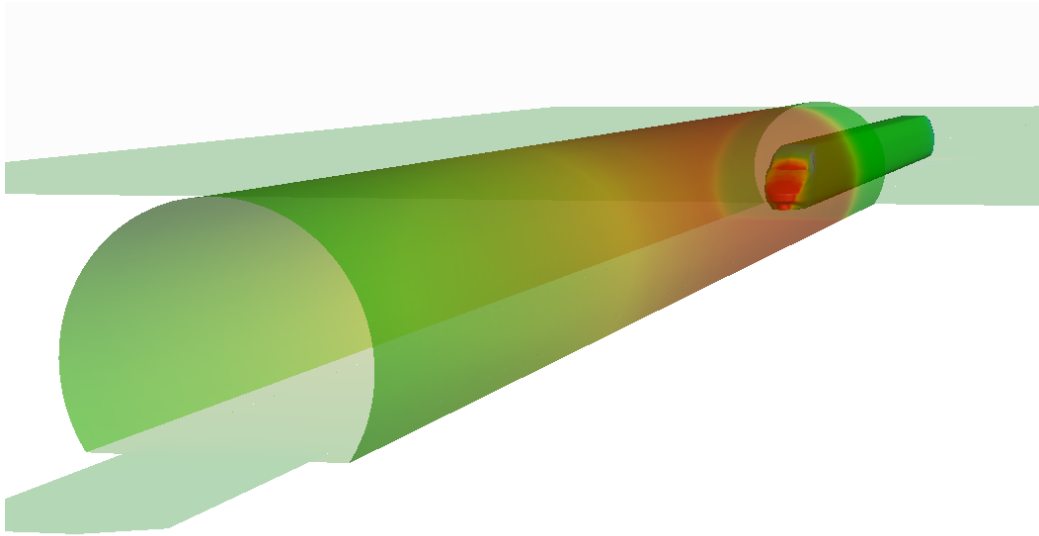
decomposed into two successive jumps: the first is consecutive to the train entry, and the second depends on the wall friction effects. Only the first is of interest. SNCF researchers defined the pressure jump value from the time corresponds to the moment where the gradient becomes constant, *i.e.* slightly after 0.2 s. Actually, this value overestimates the real value and this overestimation can be several decades of Pascals. The computation of the SNCF code slightly underestimates this value of the maximum of pressure, while our code overestimates it. On the other hand the maximum of gradient is strongly overestimated by Tg-flo (11%) and Metas1 (14%). This more important error, for our calculation, can be explained by the difference of the space mesh size.

Metas2 implies an underestimation of the pressure amplitude, but the error on the gradient of pressure is divided by more than 3 to falls to 4% and becomes clearly better than the result obtained by Tg-flo code. As the graph of the gradient shows, the curve given by the second simulation follows the shape of the curve of the experimental solution up to the end of the simulation.

Figure 20 displays pressure isocontours on the train and on the tunnel wall. When the train is at the tunnel entry, the air inside the tunnel is at rest again. After a distance of 15 m, the generation and the propagation of the compression wave are clearly highlighted. The little rectangular piece in front of the tunnel is the "*upstream zone*", previously mentioned, needful for the cyclic regeneration of cells.



(b) train at the entry



(c) train 15 m inside the tunnel

Figure 20: Generation and propagation of the compression wave (in Pa).

5. Conclusion

An Eulerian code on Cartesian meshes was developed to simulate the entrance of a train into a tunnel. The technique of sliding mesh is adopted with a conservative treatment at the faces between the sliding and fixed meshes. To accelerate the convergence towards a stationary state of the flow before the train goes into the tunnel, non-reflective boundary conditions have been implemented. This allowed to reduce the domain of calculation and also to prevent from the reflection of numerical disturbances.

The method is validated on model tests as well as on measurements *in situ*. The results show that this approach is well adapted for the simulation of the entrance of a train into a tunnel. With this methodology validated, parametrical studies are in progress to study the influence of a hood adjunction on the temporal gradient of the initial wave.

References

- [1] S. Ozawa, T. Maeda, T. Matsumura, K. Uchida, Effect of ballast on pressure wave propagating through tunnel, Proceedings of the International Conference on Speedup Technology for Railway and Maglev Vehicles 2 (1993) 299–304.
- [2] G. Matschke, C. Heine, Full scale tests on pressure wave effects in tunnels, TRANSAERO - A European initiative on transient aerodynamics for railway system optimisation, results of the Brite/Euram project Transient aerodynamics for railway system optimisation (2002) 187–195.

- [3] T. Maeda, T. Matsumura, M. Ida, K. Uchida, Effect of shape of train nose on compression wave generated by train entering tunnel, Proceedings of the International Conference on Speedup Technology for Railway and Maglev Vehicles 2 (1993) 315–319.
- [4] T. Ogawa, K. Fujii, Numerical investigation of three-dimensional compressible flows induced by a train moving into a tunnel, Computers & Fluids 26 (1997) 565–585.
- [5] C.-H. Shin, W.-G. Park, Numerical study of flow characteristics of the high speed train entering into a tunnel, Mechanics Research Communications 30 (2003) 287–296.
- [6] J.-M. Réty, R. Grégoire, Numerical simulation of the pressure wave generated by a train enters a tunnel, TRANSAERO - A European initiative on transient aerodynamics for railway system optimisation, results of the Brite/Euram project Transient aerodynamics for railway system optimisation (2002) 225–238.
- [7] J.-M. Réty, R. Grégoire, Numerical investigation of tunnels extensions attenuating the pressure gradient generated by a train entering a tunnel, TRANSAERO - A European initiative on transient aerodynamics for railway system optimisation, results of the Brite/Euram project Transient aerodynamics for railway system optimisation (2002) 225–238.
- [8] J. Mok, J. Yoo, Numerical study on high speed train and tunnel hood interaction, Journal of Wind Engineering and Industrial Aerodynamics 89 (2001) 17–29.

- [9] T. S. Yoon, S. Lee, J. H. Hwang, D. H. Lee, Prediction and validation on the sonic boom by a high-speed train entering a tunnel, *Journal of Sound and Vibration* 247(2) (2001) 195–211.
- [10] T. Hara, Aerodynamics problem when a train running into a tunnel with a large velocity, *Railway Technical research report* 153 (1960) 1–20.
- [11] M. Giles, Non-reflecting boundary conditions for Euler equation calculations, *AIAA Journal* 28(12) (1990) 2050.
- [12] F. Waymel, F. Monnoyer, M. J.-P. William-Louis, Numerical simulation of the unsteady three-dimensional flow in confined domains crossed by moving bodies, *Computers & Fluids* 35 (2006) 525–543.
- [13] F. Deister, F. Waymel, E. Hirschel, F. Monnoyer, Self-organizing hybrid cartesian grid generation and application to external and internal flows, *Numerical Flow Simulation III*. In: Hirschel EH, editor. *Note on numerical fluid mechanics and multidisciplinary design* 82 (2002) 18–29.
- [14] P. Roe, Approximate riemann solvers, parameter vectors, and difference schemes, *Journal of Computational Physics* 43 (1981) 357–372.
- [15] C. Hirsch, *Numerical Computation of Internal and external Flows, Vol II: Computational Methods for Inviscid and Viscous Flows*, Éd. wiley, new york Edition, 1997.
- [16] B. Van Leer, Towards the ultimate conservative difference scheme v.a. second order sequel to godunov’s method, *SIAM J. Sci. Stat. Comput.* 32 (1979) 101–136.

- [17] T. Barth, D. Jespersen, The design and application of upwind schemes on unstructured meshes, AIAA Paper 89-0366.
- [18] I. Sutherland, G. Hodgman, Reentrant polygon clipping, Communication of the ACM 17 (1974) 151–153.
- [19] F. Nicoud, Defining wave amplitude in characteristic boundary conditions, Journal of Computational Physics 149 (1999) 418–422.
- [20] F. Nicoud, On the amplitude of waves in characteristic boundary conditions for Euler conditions, Technical Report TR/CFD/98/21, CERFACS.