ICCFD13-2026-116

Numerical Investigation of High-Pressure Gas Expansion in Shattered Pellet Injectors for Fusion Applications

Á. Gyenge[*,**], B. Dercsényi[***] and R. Mészáros[***]
Corresponding author: gyenge.akos@ek.hun-ren.hu
[*] HUN-REN Centre for Energy Research, Hungary.
[**] Budapest University of Technology and Economics, Hungary.
[***] Eötvös Loránd University, Hungary.

Abstract: Accurate numerical modelling of propellant gas expansion in Shattered Pellet Injectors (SPI) for fusion disruption mitigation is challenging due to the extreme pressure ratio between the propellant gas (∼80 bar) and the evacuated suppressor volume (∼10^{-7} mbar). Conventional finite-volume CFD tools such as ANSYS Fluent require an artificially elevated background pressure to avoid numerical instability, introducing non-physical shock waves and limiting solution accuracy. In this work, we present a finite-volume method for simulating gas expansion into vacuum by explicitly tracking the gas-vacuum interface, using local Riemann-vacuum solutions. The approach is based on the analytical solution of the one-dimensional Riemann-vacuum problem and extends the corresponding front-tracking methodology to higher dimensions using a piecewise-linear interface representation. By restricting CFD calculations to regions where the continuum assumption remains valid, vacuum cells can be excluded from the numerical solution. The method is first validated against the analytical solution of the one-dimensional Riemann-vacuum problem, demonstrating good agreement and convergence with increasing mesh resolution. The proposed solver reproduces the physical absence of shock wave formation at the gas–vacuum interface and resolves pressures several orders of magnitude lower than Fluent's background-pressure-limited approach, while completing the 2.5 ms test case in approximately 10 seconds compared to one hour for Fluent.

Keywords: Riemann-vacuum problem, front tracking, vacuum, Finite-volume method, fusion

1. Introduction

Fusion energy production offers a sustainable, green and competitive solution to the environmental, geopolitical and economic challenges of our times. While the promises are daring, when it comes to fusion technology, most of us tend to think of delays and rising costs due to the constant prolongation of large-scale projects, such as the International Thermonuclear Experimental Reactor (ITER). The project was continuously delayed as new, unknown challenges arose during the design of this large fusion experiment.

One of these physical-technological challenges is mitigating plasma disruptions in the reactor. In tokamak-type fusion reactors, so-called plasma disruptions can occur. These instabilities lead to thermal and electromagnetic plasma quenching, which can generate high-energy runaway electron beams. While these beams do not pose a radiation threat, they can cause serious machine damage, therefore, their formation should be prevented.

In the current ITER baseline, disruption mitigation is done with Shattered Pellet Injection (SPI) [1]. The HUN-REN Centre for Energy Research (CER) Fusion Plasma Physics Department built and commissioned a test laboratory to help the development of ITER's SPI for its Disruption Mitigation System [2, 3]. Figure 1 shows the first section of the SPI device built at HUN-REN Centre for Energy Research. It includes the cryostat, where the protium, deuterium, or neon pellet is formed, the eddy-current actuated fast valve responsible for launching the pellet, the gas retention system and the Optical Pellet Diagnostics

(OPD) system. The device furthermore consists of a flight tube and a shattering head, which ultimately ends in an analysis chamber (which is only present in the test device, not included in the actual pellet injector for ITER). The total length of the SPI system is 6 m. The 27 injectors placed around the torus will form 28.5 mm-diameter cryogenic solid hydrogen pellets and launch them towards a shattering head at several hundred meters per second. The pellets are launched with single- or two-stage pneumatic gas guns, or fast valves [4, 5, 6]. To effectively mitigate disruption, it is crucial to minimise the amount of propellant gas that reaches the plasma before the solid pellet fragments, as this gas would only trigger the disruption rather than mitigate it.

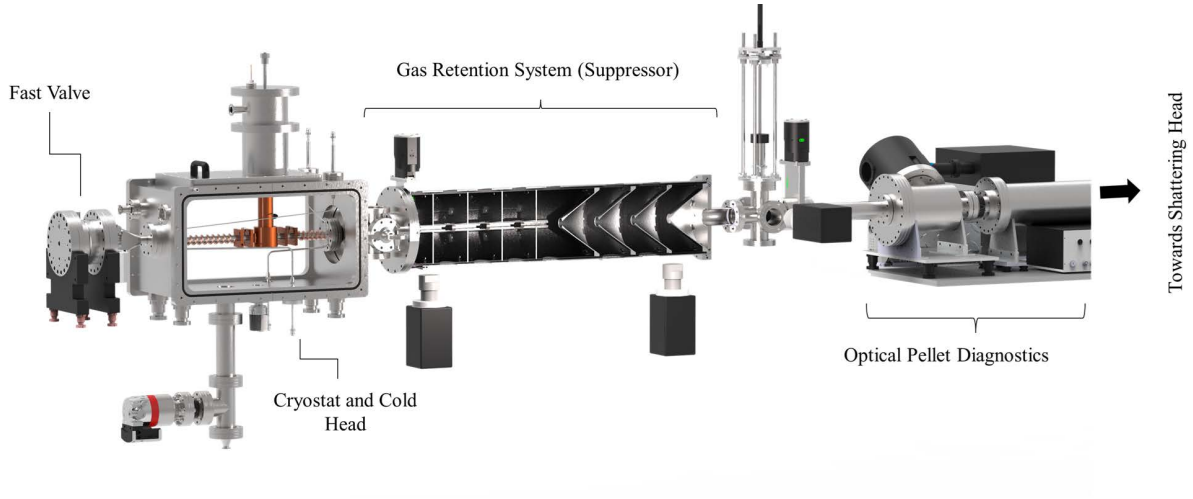


Figure 1: First half of the SPI test laboratory device at HUN-REN Centre for Energy Research with the initial suppressor design.

The SPI system employs a gas-retention or suppressor system that has undergone several recent design updates [7]. Initially, the suppressor had a single volume, with flat and conical separator elements as shown in fig. 1. During testing, several issues arose with this configuration. In general, the suppressor should satisfy the following conditions:

- It should allow the pellet to pass through undamaged, losing minimal material due to friction and collisions.
- It should ensure that the pellet is not displaced, nor tilted due to any collisions with internal elements, or the surrounding gas pressure.
- It should retain as much gas as possible: ensure that the propellant behind the moving pellet does not overtake it until the solid reaches the shattering head.

Following a series of tests and several design iterations, the suppressor design for ITER consists of two volumes, separated by a gate valve. The first section (Suppressor A) has three cylindrical compartments connected by four guiding rails, and the second (Suppressor B) has four conical separators with a 40 mm opening [7]. The cross-section of the suppressor version used in the test laboratory is shown in fig. 2

The suppressor design was assisted by Computational Fluid Dynamics (CFD) simulations; however, accurately modelling the problem is challenging with traditional numerical methods. The volume is initially evacuated to 10^{-7} mbar, while the propellant gas pressure is around 80 bar. A significant portion of the domain operates in the rarefied regime, with Knudsen numbers well above unity, making classical finite-volume schemes unreliable in these regions, particularly for modelling the expansion front. Meanwhile, the high-pressure regions and strong temporal dependence make kinetic-theory-based methods computationally inefficient. The pellet launch process takes up around 20 ms, during which the propellant gas undergoes extreme expansion and reaches velocities up to six times its local speed of sound (Mach 6). Therefore, simulating this process requires a fast and robust numerical method.

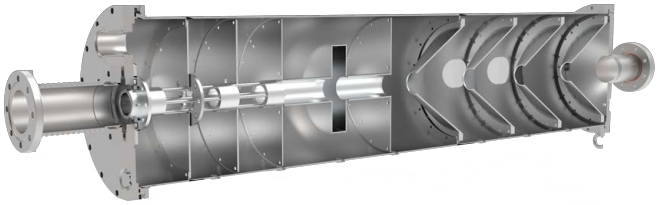


Figure 2: Cross-section of the final design of the propellant gas suppressor used for the SPI test lab.

2. Problem Statement

According to the current ITER design methods, the suppressor efficiency is analysed with commercially available CFD tools (ANSYS Fluent). As this software is incapable of treating a perfect vacuum and produces high numerical errors and is prone to instability at very low pressures, the background pressure is set to 100 Pa, which is 7 orders of magnitude higher than the actual operational pressure. In our research, we investigate the applicability of a different finite-volume-based approach for numerically predicting gas behaviour in the SPI's suppressor system.

In our approach, we treat the expansion front propagation as a Riemann (more specifically, Riemann-vacuum) problem (RVP). In one spatial dimension, the Riemann problem is an initial-value problem in which a 1-dimensional space is divided by a separating membrane; on either side of it, gases are at rest in different states (the species, pressure, momentum and temperature cannot all be equal on the two sides of the diaphragm). The Riemann-vacuum problem is the limit of this , where on one side of the membrane, there is a perfect vacuum. The RVP has an exact solution in 1D, and several numerical methods were developed to approximate this result [8]. Our code solves the discretized Euler equations with finite-volume method. In our study, we neglect the effects of viscosity and friction for the first approximation. The conservation equations are given as

$$\frac{\partial \mathbf{u}}{\partial t} + f(\mathbf{u}) = 0, \quad \text{with} \quad \mathbf{u} = \begin{pmatrix} \rho \\ \rho v \\ e \end{pmatrix} \quad \text{and} \quad f(\mathbf{u}) = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ v(e+p) \end{pmatrix}, \tag{1}$$

with ρ, v and e being the density, velocity, and internal energy of the gas, respectively. For the RVP, the initial values are $(\rho_0, m_0, e_0)^T$ for $x < 0$ and $(0, 0, 0)^T$ for $x > 0$. In his paper, C.-D. Munz proposes an algorithm to track the gas-vacuum boundary in one dimension [8]. At the gas-vacuum boundary, he introduces a new flux definition, by averaging the exact solution of the RVP over a time interval,

$$\mathbf{g} = \frac{\sigma}{\sigma - \min(0,\, v - c)} \left(f(\mathbf{u}) - \min(0,\, v - c)\mathbf{u} \right), \tag{2}$$

with σ, v and c being the velocity of the gas-vacuum boundary, the flow velocity and the local speed of sound, respectively. While the approximation is consistent with the entropy inequality (i.e. the second law of thermodynamics), the calculation method for the speed of sound does have an effect on the result, due to discrete numerics. The speed of sound is defined as $c^2 = \left.\frac{\partial p}{\partial \rho}\right|_s$, which for ideal gases, reads as $c(p, \rho) = \sqrt{\gamma \frac{p}{\rho}}$. For an isentropic case, such as the free expansion of gas into vacuum, this can be simplified to a definition only dependent on density, $c(\rho) = \sqrt{\gamma K \rho^{\gamma - 1}}$, where K is a constant. Depending on which formulation is used, the 1D code yields different results. Figure 3 shows the analytical solution and the result of two simulations for standard state Oxygen gas in a 1 m long tube at $t = 0.3$ ms.

As the figure shows, none of the simulations returns precisely the location of the expansion front (i.e. the location of the first cells, where the field variables are non-zero). This does not mean that the methods have a large error: the density, pressure, momentum and energy have very small errors near the expansion

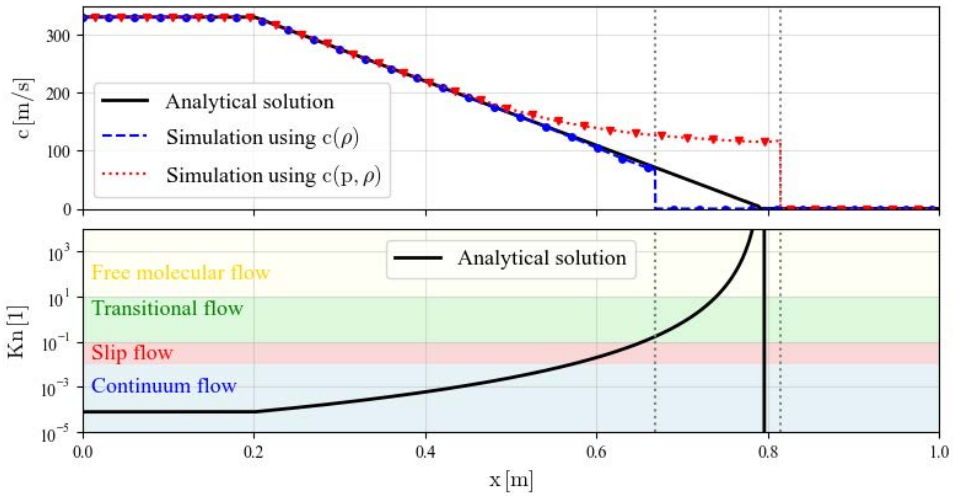


Figure 3: Comparison of the analytical 1D solution and two numerical results with different sound speed formulation for standard state oxygen at $t = 0.3$ ms (above) and the cell Knudsen number in that moment (below).

front, as shown in Munz's original paper [8], but the absolute values are so small that numerical precision dictates the accuracy. In our code, we use the isentropic formulation for the reasons detailed below. In the figure, the bottom graph shows the local Knudsen number (Kn) in the simulation domain, defined as

$$\mathrm{Kn} = \frac{\lambda}{L} = \frac{k_B}{\sqrt{2}\pi d^2 R}\frac{1}{\rho L}, \tag{3}$$

where λ is the mean free path, d is the diameter of the gas particles, L is the system's characteristic length (taken as $L = \Delta x$ here). As seen in fig. 3, the numerical results for the isentropic speed of sound only deviate from the analytical results where $\mathrm{Kn} > 0.1$, that is, where the flow is not a continuum any more. This highlights an important limitation of our approach: we do not want to create a solver that handles both the rarefied and continuum regimes of the domain. Since we are solving the Euler equations, which are based on the continuum assumption, we can only handle the low-Knudsen regions. In our work, we focus on identifying the region where classical CFD solvers can be effectively used without special treatment for low-pressure cells, where the solution is not computed. Our goal is to develop a numerical code capable of tracking the evolution of the gas-vacuum boundary in higher dimensions using the 1D solution of the Riemann-vacuum problem.

3. Methods

In our work, we extend the tracking algorithm to higher dimensions to model the propellant gas expansion within the SPI system. This allows the exclusion of vacuum cells from the calculations, improving stability and applicability. This process requires a technique to geometrically track the evolution of the gas-vacuum boundary in space, which is achieved using a piecewise-linear tracking method. We define a set of boundary points in the domain and update their positions over time according to the constitutive equations. This enables interface tracking with precision below the cell resolution.

3.1 Front tracking

In two dimensions, the gas–vacuum interface is represented by a piecewise-linear curve defined by a finite set of marker points $\mathbf{q}_k$. The interface geometry is reconstructed by linear interpolation between neighbouring points, its time evolution reduces to tracking the motion of these segments. At each time step, the new position of each line is calculated from local RVPs solved for each segment, from which

the new positions of marker points are obtained. Thus, the two-dimensional problem can be decomposed into a finite set of one-dimensional RVPs.

The velocity of the gas-vacuum interface can be determined using the state of the gas behind the front. The following expression can be derived using the methods of characteristics [9]:

$$\sigma = v + \frac{2c}{\gamma - 1}, \tag{4}$$

where the velocity v is the velocity component perpendicular to the front. To define the one-dimensional RVP and therefore get the velocity of the gas-vacuum interface for a segment, we need the state vector behind that segment. Because the segment may cross cell boundaries, we introduce a so-called *synthetic* cell.

We construct a rectangular synthetic cell whose two adjacent corners are the marker points $\mathbf{q}_k$ and $\mathbf{q}_{k+1}$, with depth Δx, as shown in fig. 4. In an axisymmetric system, this rectangle is the cross-section of the synthetic cell at a given azimuthal angle φ. The synthetic cell overlaps with the computational grid, and we determine its state vector $\mathbf{u}_s$ from the overlapping cells. The overlap area with cell I is A_I, calculated using grid sampling: the synthetic cell is divided into a uniform sub-grid, and A_I is estimated by counting sub-grid points falling inside cell I, weighted by the area per point. In the axisymmetric case, each sub-grid point is additionally weighted by its radius.

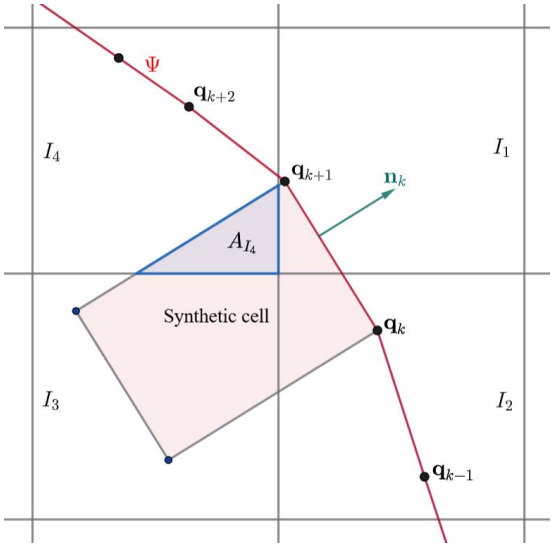


Figure 4: Synthetic cell behind the segment connecting $\mathbf{q}_k$ and $\mathbf{q}_{k+1}$. The normal vector $\mathbf{n}_k$, and the area of overlap with the I_4-th cell are shown.

In the finite volume method, the state vector $\mathbf{u}_I$ is stored under the assumption that cell I is completely filled with gas. We correct for this using the gas volume fraction $\bar{V}_I \in [0, 1]$, which equals 1 in fully filled cells and 0 in vacuum. The state vector of the synthetic cell is then:

$$\mathbf{u}_s = \frac{1}{V_s} \int_{V_s} \mathbf{u}\, dV = \sum_I \frac{1}{\bar{V}_I} \left(\frac{A_I}{V_s} \right) \mathbf{u}_I, \tag{5}$$

where V_s is the volume of the synthetic cell. From $\mathbf{u}_s$ we extract the velocity and speed of sound, which are then used in eq. (4) to obtain the interface velocity.

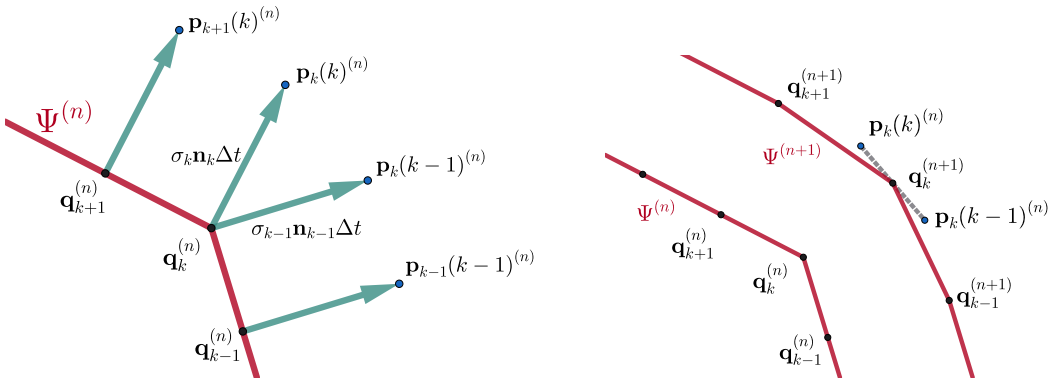
Each segment has its own interface velocity σ_k and outward normal $\mathbf{n}_k$, computed from the synthetic cell between $\mathbf{q}_k$ and $\mathbf{q}_{k+1}$. To advance marker point $\mathbf{q}_k$, we propagate it along the normals of both adjacent

segments to obtain temporary points $\mathbf{p}_k(k)^{(n)}$ and $\mathbf{p}_k(k-1)^{(n)}$ and then take the average:

$$\mathbf{p}_k(k)^{(n)} = \mathbf{q}_k^{(n)} + \sigma_k \mathbf{n}_k \Delta t, \quad \mathbf{p}_k(k-1)^{(n)} = \mathbf{q}_k^{(n)} + \sigma_{k-1}\mathbf{n}_{k-1}\Delta t. \tag{6}$$

$$\mathbf{q}_k^{(n+1)} = \frac{1}{2}\left(\mathbf{p}_k(k)^{(n)} + \mathbf{p}_k(k-1)^{(n)}\right). \tag{7}$$

This is illustrated in fig. 5.



(a) Temporary points (b) New marker points

Figure 5: Time evolution of the marker points. (a) shows the creation of the temporary **p** points, and (b) shows the new marker points and the new interface.

This keeps the number of marker points constant, thus increasing the distance between them during gas expansion and decreasing the precision of the interface. To prevent this, we introduced a threshold: if the distance between adjacent marker points exceeds this, a new marker point is created at their midpoint. In practice, we set this threshold to Δx, thus in each cell with a gas-vacuum interface, there is at least one marker point, and each segment ends within neighbouring cells. If a marker point is on the wall of the system, it moves along that wall. It is achieved by zeroing out the component of $\sigma\mathbf{n}$ perpendicular to the wall.

To calculate the $\bar{V}_I$ ratio, we first need to introduce auxiliary variables describing the position of the gas-vacuum interface. Let $\alpha(I) \in [0,1]$ and $\beta(I) \in [0,1]$ denote the normalised position of the gas-vacuum interface on the upper and right cell boundaries of cell I. Let furthermore J_3 and J_4 denote the cells below and to the left of cell I. Using the $\alpha(I)$, $\beta(I)$, $\alpha(J_3)$, and $\beta(J_4)$ interface positions, the $\bar{V}_I$ volume fraction can be calculated. For computational efficiency, we implemented a linear approximation; during the evaluation of $\bar{V}_I$, we assume the segment of the gas-vacuum interface inside the cell to be linear. This allows the calculation to be done solely from the α and β variables. The method is visualised in fig. 6.

Similarly to $\alpha(I)$ and $\beta(I)$, let $\alpha_0(I)$ and $\beta_0(I)$ denote the normalized position of the gas-vacuum interface on the lower and left cell boundaries in the previous time step. Equivalently:

$$\alpha_0^{(n)}(I) = \alpha^{(n-1)}(J_3), \qquad \beta_0^{(n)}(I) = \beta^{(n-1)}(J_4). \tag{8}$$

Where J_3 and J_4 denote the cells below and to the left of cell I, respectively. The total numerical flux can be evaluated by considering two distinguished flux contributions; the gas-gas flux, introduced in section 3.2, and the gas-vacuum flux defined in eq. (2). The total flux can be obtained by weighting these contributions with the associated cell boundary ratios, determined by the previously introduced α, β, α_0, β_0 variables, as shown in fig. 6.

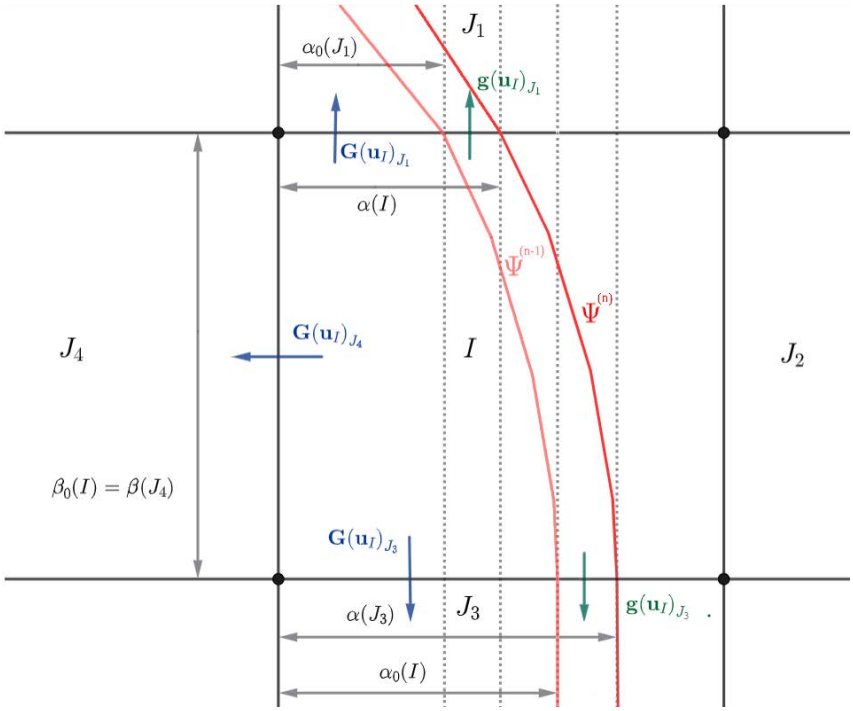


Figure 6: Schematic figure of cell I, illustrating the flux calculation.

3.2 HLL method

In our code, when two neighbouring cells are filled with gas, fluxes through their shared face are computed by a Harten-Lax-van Leer (HLL) Riemann solver [10]. Let $\mathbf{u}_L^{(n)}$ and $\mathbf{u}_R^{(n)}$ denote the states on the left and right sides of the cell interface at time step n. The numerical flux $\mathbf{G}^{(n)}$ is given by:

$$\mathbf{G}^{(n)} = \begin{cases} f(\mathbf{u}_L^{(n)}), & S_L^{(n)} \geq 0, \\ \dfrac{S_R^{(n)} f(\mathbf{u}_L^{(n)}) - S_L^{(n)} f(\mathbf{u}_R^{(n)}) + S_L^{(n)} S_R^{(n)} \left(\mathbf{u}_R^{(n)} - \mathbf{u}_L^{(n)}\right)}{S_R^{(n)} - S_L^{(n)}}, & S_L^{(n)} < 0 < S_R^{(n)}, \\ f(\mathbf{u}_R^{(n)}), & S_R^{(n)} \leq 0, \end{cases} \tag{9}$$

where $S_L^{(n)}$ and $S_R^{(n)}$ are the signal velocities of the left and right outer waves respectively. This first order explicit scheme does require an approximation for the aforementioned S signal velocities. In our implementation, we used both the Davis signal velocity [9]:

$$S_L = \min\left(v_L - c_L,\, v_R - c_R\right), \qquad S_R = \max\left(v_L + c_L,\, v_R + c_R\right)$$

and the approximation introduced by Bernd Einfeldt [11]. For low-resolution, 2- or higher-dimensional simulations, we observed slightly different behaviours with the two signal velocity approximations. This is likely due to signal velocity's effect on the numerical dissipation that the HLL method introduced to the system.

The overall flux of a cell is the sum of fluxes on the cell boundaries, where it can be computed with the fluxes $\mathbf{G}$ (eq. (9)) and $\mathbf{g}$ (eq. (2)) over their associated area F:

$$F_J(g) = \begin{cases} \Delta x(\alpha - \alpha_0) & \text{for horizontal wall} \\ \Delta y(\beta - \beta_0) & \text{for vertical wall} \end{cases} \qquad F_J(G) = \begin{cases} \Delta x \alpha_0 & \text{for horizontal wall} \\ \Delta y \beta_0 & \text{for vertical wall} \end{cases}, \tag{10}$$

where $F_J(g)$ is the area of gas-vacuum flux on the wall between the I-th and J-th cell, while $F_J(G)$ is

the area on the same wall for the gas-gas flux. Combining it all, we get the advection equation:

$$\mathbf{u}_I^{(n+1)} = \mathbf{u}_I^{(n)} - \frac{\Delta t}{V_I} \sum_J \left[F_J(g)\mathbf{g}^{(n)} + F_J(G)\mathbf{G}^{(n)} \right]. \tag{11}$$

4. Results and Future Work

To verify the implementation, we validate the code against analytical solutions for one-dimensional and quasi-one-dimensional test cases. We also compare our results with those obtained using the commercial CFD solver ANSYS Fluent, which is currently employed by ITER for suppressor design studies.

4.1 One dimensional validation

The one-dimensional Riemann vacuum problem has a known analytical solution [8, 9], which provides a suitable benchmark for code validation. To model this problem, we use a two-dimensional rectangular mesh over the domain $0\,\mathrm{m} \le x \le 1\,\mathrm{m}$, and $0\,\mathrm{m} \le y \le 5\Delta y$, with $\Delta y = \Delta x$. The mesh resolution is defined by the number of cells in each direction. For all one-dimensional validation cases, the number of cells in the y direction is fixed at $N_y = 5$, while the number of cells in the x direction, N_x, is varied. The cell size is therefore $\Delta x = \frac{1\,\mathrm{m}}{N_x}$. Standard nitrogen gas is used with the following initial conditions:

$$\mathbf{u}(t=0) = \begin{cases} (1.123\,\mathrm{kg/m^3}, 0, 0, 1\,\mathrm{bar}) & \text{for} \quad x \le 0.3\,\mathrm{m} \\ (0,0,0,0) & \text{elsewhere} \end{cases}. \tag{12}$$

We monitored the pressure at $x = 0.5\,\mathrm{m}$, yielding p_i pressure values. For the validation process, we used the root mean square error:

$$\sqrt{\frac{1}{N_t} \sum_{i=1}^{N_t} (p_i - f(t_i))^2}, \tag{13}$$

where N_t is the number of time steps, and $f(t)$ is the analytical solution.

The resulting pressure errors and simulation times are summarised in table 1. Since the test problem is inherently one-dimensional, the solution should remain uniform in the y direction. To verify this, the pressure was sampled at $x = 0.5\,\mathrm{m}$ in each of the five cells along the y direction. For non-zero pressure values, the relative error between the samples was computed; the results are shown in fig. 7.

Figure 8b compares the analytical solution with the $N_x = 2000$ simulation and the low-pressure ANSYS Fluent solution at $x = 0.5\,\mathrm{m}$. The deviation from the analytical solution is shown in fig. 8a for the three mesh resolutions considered, together with the pressure-based ANSYS Fluent simulation for comparison.

The results demonstrate that the proposed method reproduces the analytical solution with high accuracy. As shown in table 1, the root-mean-square error decreases consistently with increasing mesh resolution, indicating numerical convergence.

The one-dimensionality of the solution is also well preserved. Although a relative deviation of up to approximately 2% can be observed between the different rows of cells when the expansion front reaches the measurement location, this occurs only over a very short time interval and at extremely low pressure levels. The corresponding maximum absolute errors are only 1.2 Pa, 0.04 Pa, 0.005 Pa for resolutions 500, 1000, 2000 respectively. These values decrease rapidly with increasing resolution and are negligible compared with the characteristic pressure scale of the problem.

4.2 Results in higher dimensions

In the absence of an analytical solution in higher dimensions, the results were compared with simulations performed using Ansys Fluent. This software is not only common in the industry, but also used for the design of the ITER suppressor system. The simulation domain was axisymmetric with the axis of

N_x	500	1000	2000
RMS error [Pa]	55.9	35.46	21.1
Computational time [s]	12	49	173

Table 1: Root mean square error of the pressure at $x = 0.5\,\mathrm{m}$ for 3 different mesh sizes, and the simulation's computational time.

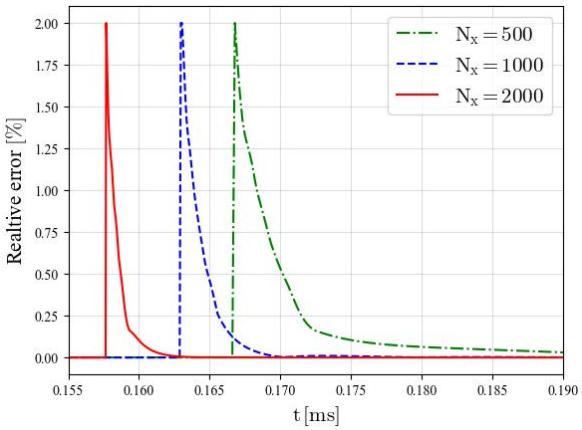
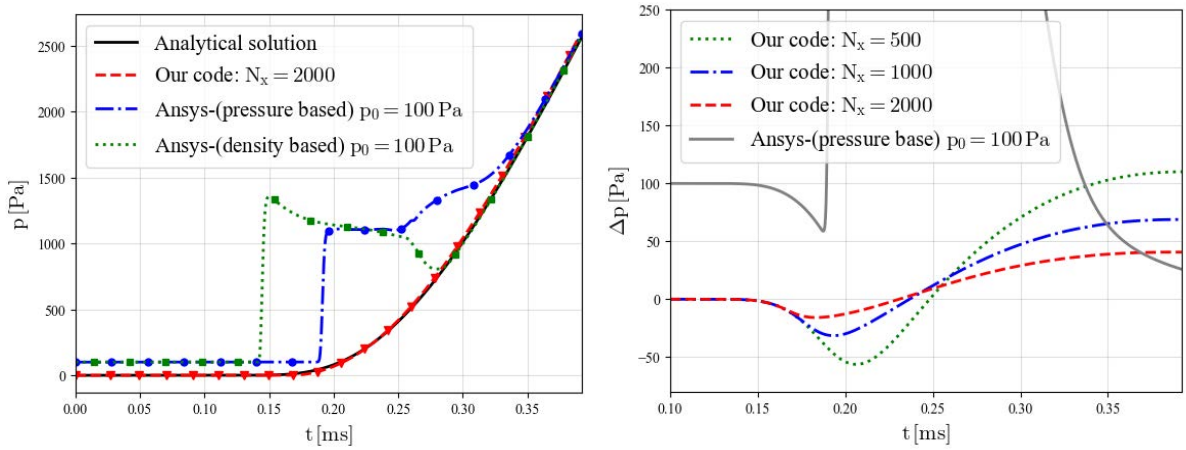


Figure 7: Relative error of the pressure for all 3 resolution. The jump happens, when the front reaches the measurement point.



(a) Pressure values

(b) Pressure difference

Figure 8: At $x = 0.5\,\mathrm{m}$ the (a) pressure values from our $N_x = 2000$ resolution simulation ($\triangledown$) from Ansys Fluent pressure based ($\circ$) and density based ($\square$) solver using $p_0 = 100\,\mathrm{Pa}$, and (b) the pressure difference from the analytical solution.

symmetry located at $r = 0$. The projection of the geometry onto the $r - z$ plane is shown in fig. 9.

The axisymmetric mesh consisted of $N_r = 15$ and $N_z = 70$ cells, corresponding to $\Delta r = \Delta z = 0.01\,\mathrm{m}$. As in the one-dimensional case, the simulations were performed with standard nitrogen gas. At $t = 0$ the gas is present in the $(r, z) \in [0,\ 0.15\,\mathrm{m}] \times [0,\ 0.15\,\mathrm{m}]$ region, as shown in fig. 9. The initial state of the gas was: $\rho = 1.123\,\mathrm{kg/m^3}$, $\mathbf{v} = 0$, $p = 1\,\mathrm{bar}$. The evolution of the gas-vacuum interface is shown in fig. 10.

For the Fluent simulations, a background pressure of 100 Pa was prescribed in the vacuum region, and the pressure-based solver was used. To compare the two approaches, the pressure was monitored at $r = 0.15\,\mathrm{m}$ and $z = 0.55\,\mathrm{m}$. The results are shown in fig. 11.

The comparison reveals similar curves; however, a few key differences appear. Most notably, the Fluent solution shows an initial pressure wave that is absent from our results. This feature is likely caused by the non-zero background pressure imposed in the Fluent model, which can induce acoustic or shock waves that are not actually present in a vacuum.

Additional differences appear during the pressure decay phase. The Fluent solution exhibits a less steep

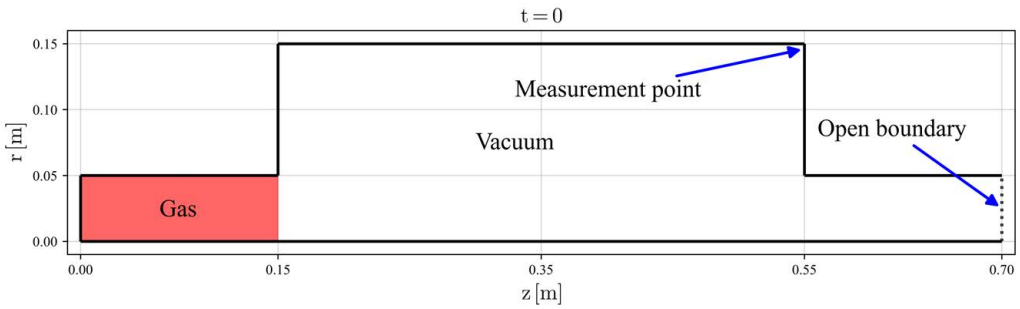


Figure 9: Schematic figure of axisymmetric system.

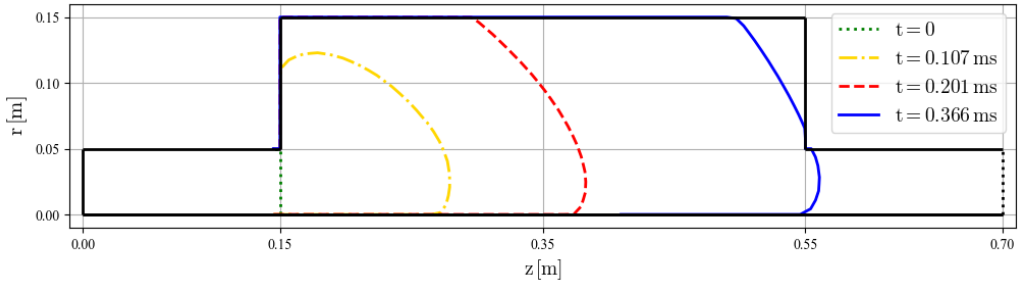


Figure 10: Position of the gas-vacuum interface at some points in time.

decrease in pressure and contains a local maximum at approximately $t = 1.8\,\text{ms}$. The less steep descent can be attributed to fundamentally different methods for flux calculation within the gas region used by the two simulations. We ran two simulations, one with Davis and one with Einfeldt signal velocity, and they produced nearly identical decay curves. While the Einfeldt solution shows somewhat better agreement with Fluent near the first pressure peak, neither simulation reproduces the secondary maximum observed in the Fluent results. It is presumably a consequence of the isentropic approximation. While the free expansion of the gas into vacuum can be considered isentropic, the gas can form reflected shock waves at the boundaries of the system; thus, the isentropic assumption can be violated. Therefore, in our simulation, the reflection shock wave will be smeared out, which may explain the absence of the secondary pressure peak in our simulations.

For further qualitative comparison, we examined the pressure distributions obtained with ANSYS Fluent (fig. 12), (fig. 13). Both figures show the pressure at $t = 0.366\,\text{ms}$ on a logarithmic scale. The minimum pressure in the ANSYS Fluent simulation is approximately 17 Pa above the prescribed background pressure. In contrast, our solver predicts pressures as low as $10^{-5}\,\text{Pa}$. However, for visualisation purposes, the lower limit of the colour scale was set to 1 Pa. Consequently, all cells with pressures below this threshold are displayed using the same colour.

Three main differences can be observed between the two solutions, aside from the presence of the background pressure in the Fluent simulation. First and most importantly, the Fluent results exhibit a distinct shock wave, which is absent from our results. This is consistent with the theoretical behaviour of a gas-vacuum interface, where a shock wave formation is not expected, as shown by T.-P. Liu and J. A. Smoller [12].

Second, the upper left corner (at (0.15 m, 0.15 m)) behaves differently. Ansys predicts relatively high pressure due to the shock wave. On the other hand, our code gives an extremely low pressure (around 0.5 Pa). Finally, the expansion front in our solution appears to be substantially farther from the initial gas volume than the shock front visible in the Fluent results. At first glance this may suggest that the gas expands significantly faster in our simulation. However, the pressure near this leading edge is on the order of $(10 \times 10^{-5}\,\text{Pa})$, indicating that only a very rarefied gas has reached this region. From the

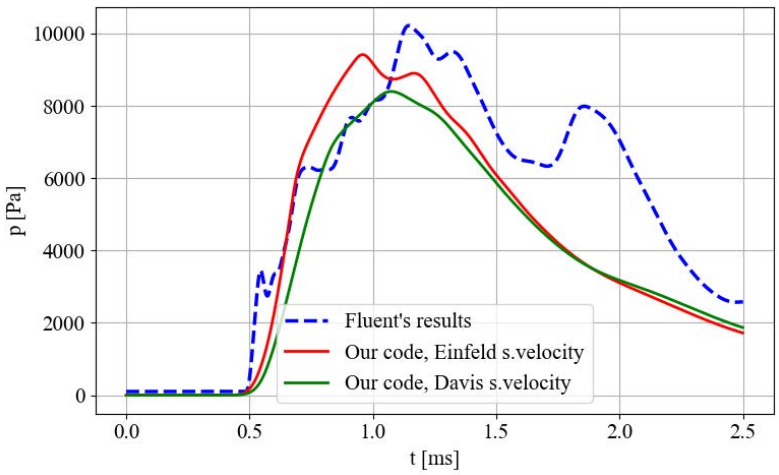


Figure 11: Comparison of our results with simulations run in Fluent.

measurement point located on the upper right corner, we can see that the disturbance reaches this location at approximately the same time in both simulations (fig. 11).

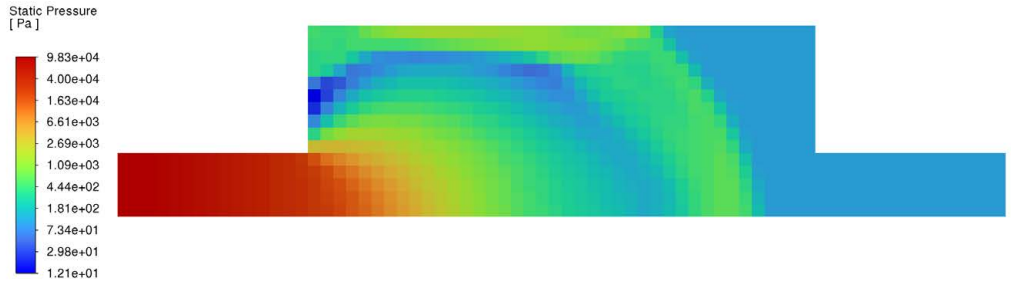


Figure 12: Ansys Fluent pressure distribution at $t = 0.366$ ms.

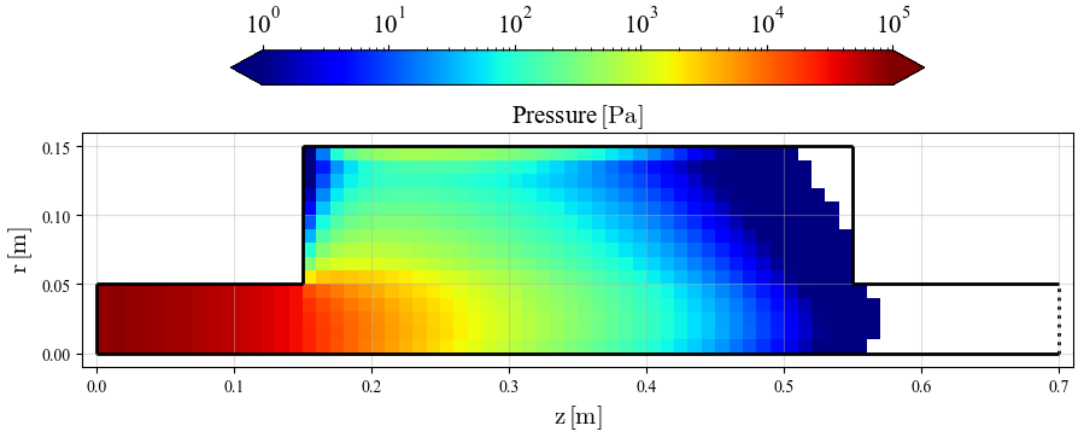


Figure 13: Our solvers pressure distribution at $t = 0.366$ ms.

4.3 Future development and experimental validation

In conclusion, the results presented in this work demonstrate good agreement with both analytical solutions and established CFD tools. For the full, 2.5 ms test case, the ANSYS Fluent simulation required approximately one hour of computation time, whereas our solver completed the same calculation in

roughly 10 seconds. However, this computational advantage is limited to relatively small problems. In its current implementation, fluxes are evaluated directly, causing the computational cost to increase rapidly with mesh size. As a result, the present code is not yet suitable for efficient simulations on the large meshes required for realistic SPI geometries.

Future development will focus on improving computational efficiency and implementing the gas-vacuum tracking algorithm within OpenFOAM. Besides significantly increasing the accessibility of the method for other researchers, this approach would allow the use of iterative solvers, advanced flux formulations, parallelisation strategies, and other tools that have been developed for compressible flow simulations by the community.

Once running on a full-scale SPI model becomes available, we can validate our results with experimental data that is already available at the HUN-REN Centre for Energy Research. Such validation is essential for assessing the accuracy of our approach and for determining if explicitly tracking the gas-vacuum boundary and performing CFD calculations only in the regions where the continuum assumption stands, can provide more reliable results for supporting the design of gas suppressors for Shattered Pellet Injectors in future fusion power plants.

Acknowledgements

Project no. 2024-2.1.2-EKÖP-KDP-2024-00005 has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the **EKÖP_KDP-24-1-BME-19** funding scheme.

The authors would also like to thank Dr. Balázs Farkas for his guidance.

References

[1] S Jachmich, M Lehnen, U Kruezi, F J Artola, M Dibon, S Giors, M Kochergin, A Loarte, T Luce, R A Pitts, L R Baylor, T Boujet, D Dunai, T E Gebhart, Á Gyenge, A Horvat, G Kocsis, J Manzagol, A Matsuyama, P Matura, F Millet, E Nardon, D I Réfy, U Sheikh, S Signetti, T Szepesi, I V Vinyar, S Zoletnik, and A Zsákai. ITER Disruption Mitigation System Design and Application Strategy. In *30th IAEA Fusion Energy Conference*, 2025.

[2] S. Meitner, L.R. Baylor, N. Commaux, D. Shiraki, S. Combs, T. Bjorholm, T. Ha, and W. McGinnis. Design and Commissioning of a Three-Barrel Shattered Pellet Injector for DIII-D Disruption Mitigation Studies. *Fusion Science and Technology*, pages 1–6, July 2017.

[3] S. Zoletnik, E. Walcz, S. Jachmich, U. Kruezi, M. Lehnen, G. Anda, T. Szabolics, T. Szepesi, G. Bartók, G. Cseh, Z. Boros, D. Dunai, G. Gárdonyi, J. Hakl, S. Hegedűs, I. Katona, A. Kovacs, G. Kocsis, M. Lengyel, S. Mészáros, D. Nagy, D. Oravecz, L. Poszovecz, D. Réfy, K. Vad, and M. Vécsei. Shattered pellet technology development in the ITER DMS test laboratory. *Fusion Engineering and Design*, 190:113701, May 2023.

[4] T.E. Gebhart, L.R. Baylor, M. Dibon, M.N. Ericson, E.J. Felske, S.S. Frank, W.L. Gardner, A.G. Ghiozzi, S. Jachmich, U. Kruezi, M. Lehnen, and D.A. Velez. Impact of breech geometry and propellant flow on the release of large pellets for the ITER disruption mitigation system. *Nuclear Fusion*, 64(3):036021, March 2024.

[5] D. Nagy and D.I. Réfy. Eddy current actuated fast valve development for disruption mitigation applications. *Fusion Engineering and Design*, 202:114400, May 2024.

[6] D. Nagy, Á. Gyenge, B. Farkas, and D.I. Réfy. Validation of a new propellant valve test bed to support developments for shattered pellet injection technology. *Fusion Engineering and Design*, 218:115142, September 2025.

[7] R.L. Csiszár, Á. Gyenge, S. Hegedűs, U. Kruezi, M. Dibon, S. Jachmich, and S. Zoletnik. Comparison of propellant gas suppressor configurations for Shattered Pellet Injector. *Fusion Engineering and Design*, 227:115742, June 2026.

[8] C.-D. Munz. A tracking method for gas flow into vacuum based on the vacuum Riemann problem. *Mathematical Methods in the Applied Sciences*, 17(8):597–612, June 1994.

[9] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. Springer, Dordrecht New York, 3rd ed edition, 2009.

[10] A. Harten, P. Lax, and B. van Leer. On upstream differencing and godunov-type schemes for hyperbolic conservation laws. *Siam Review*, 25:35–61, January 1983.

[11] B. Einfeldt. On Godunov-Type Methods for Gas Dynamics. *SIAM Journal on Numerical Analysis*, 25(2):294–318, April 1988.

[12] T. P Liu and J. A Smoller. On the vacuum state for the isentropic gas dynamics equations. *Advances in Applied Mathematics*, 1(4):345–359, December 1980.