Continuum-based Particle Gas (CPG): A New Approach for Airbag Deployment Simulations

Edouard Yreux, Jason Wang, Iñaki Caldichoury, Mohammed Mujtaba Atif

Ansys, Inc

1 Background of Airbag Deployment Simulations

The evolution of automotive safety systems has witnessed a remarkable journey over the past few decades, with airbags emerging as pivotal components in mitigating the severity of injuries during vehicle collisions. Initially conceived as relatively simple passive restraint systems, airbags have undergone a profound complexification in their design and functionality, driven by the relentless pursuit of enhanced occupant protection and regulatory compliance. Today, modern vehicles incorporate a diverse array of airbags strategically positioned throughout the cabin to address various collision scenarios. From front and side airbags to curtain and knee airbags, this proliferation underscores the nuanced approach to occupant protection adopted by automotive manufacturers.

However, from a numerical analysis standpoint, this increased complexity has introduced new challenges. Modeling airbag deployment has been difficult from the outset due to the intricate dynamics of airbag inflation and the complex Fluid-Structure-Interaction (FSI) involving gas, airbag fabric, and internal components. While initial endeavors primarily aimed to accurately depict the interaction between occupants and fully inflated airbags, modern CAE tools must now also predict the entire deployment phase with exceptional precision, necessitating the incorporation of complex physics into the numerical methods.

Early modeling efforts relied on a Control Volume (also called Uniform Pressure) approach where explicit representation of the gas is replaced with a simple uniform pressure model. While the fully deployed geometry is correct, deployment kinematics is notoriously inaccurate. Developed two decades later, the Corpuscular Particle Method (CPM), rooted in kinetic gas theory, quickly emerged as the preferred technique for sophisticated airbag modeling. This method adeptly handles intricate airbag designs and has demonstrated considerable utility. However, the recent trend towards increasingly complex airbag designs, coupled with the growing need for high-fidelity resolution during initial deployment stages, has underscored certain shortcomings in effectively resolving local flow characteristics using this method. CPM involves numerous numerical parameters that require calibration and tuning to align with experimental tests, which presents challenges in two aspects:

1. Due to the absence of predictive capabilities, airbag design continues to rely on experimental tests, with numerical models being developed retrospectively and calibrated to match these tests. Consequently, simulation cannot be utilized to make design choices in the airbag manufacturing process. This is in sharp contrast to other areas of vehicle design, where predictive numerical simulations are used extensively to inform a wide range of design decisions before the vehicle is constructed.
2. In crashworthiness analysis, numerical airbag models must be meticulously calibrated to align with experimental data. Achieving accurate correlation across all load cases (dummy sizes and positioning) can be difficult, often necessitating the tuning of different models for various scenarios.

To address these concerns, we propose a novel method for simulating airbag deployment known as Continuum-based Particle Gas (CPG), which relies on continuum physics principles. Like CPM, CPG is a particle-based approach that eliminates the need for meshing the airbag's internal volume. However, CPG adopts continuum theory and resolves the compressible Navier-Stokes equation coupled with an ideal gas equation of state. CPG aims to provide precise gas dynamics and eliminate the need for numerical parameter calibration to achieve accurate model correlation. The objective is to elevate the predictive capabilities of airbag deployment simulations to the same standard as other aspects of crash analysis, enabling users to rely on numerical results prior to hardware testing.

© 2024 Copyright by Ansys Inc.

2 CPG Theory

CPG is a particle method based on a generalized finite difference framework. The domain Ω is represented by a collection of particles that sample the interior volume as well as the outer and internal surfaces. Each particle is equipped with a kernel function ϕ_h of compact support h (for example, a cubic-B-Spline), and the approximation space is constructed using linear Moving-Least-Squares (MLS) approximation functions $\{\Psi_I\}_I$. For any point $\boldsymbol{x}$ in the domain Ω, a neighbor list $\{J\}$ is constructed containing all the particles whose support h cover the evaluation point. For non-convex domains, special care must be taken to ensure neighbors don't interact through the domain boundary or through internal surfaces. The approximation function associated with particle I evaluated at point $\boldsymbol{x}$ then reads [1]:

$$\Psi_I(\boldsymbol{x}) = \boldsymbol{H}^T(\boldsymbol{0})\boldsymbol{M}^{-1}(\boldsymbol{x})\boldsymbol{H}(\boldsymbol{x}-\boldsymbol{x_I})\,\phi_h(\boldsymbol{x}-\boldsymbol{x}_I)$$

with $\boldsymbol{H}^{\mathrm{T}}(\boldsymbol{x})$ =[1 x y z] and $\boldsymbol{M}(\boldsymbol{x}) = \sum_J \boldsymbol{H}(\boldsymbol{x}-\boldsymbol{x}_J)\boldsymbol{H}^T(\boldsymbol{x}-\boldsymbol{x}_J)\phi_h(\boldsymbol{x}-\boldsymbol{x}_J)$.
Our conservative variables read as

$$\boldsymbol{U} = \begin{Bmatrix} \rho \\ \rho u_x \\ \rho u_y \\ \rho u_z \\ \rho E \\ \rho c_1 \\ \vdots \\ \rho c_{n_{sp}} \end{Bmatrix}$$

where ρ is the fluid density, $\boldsymbol{u} = \{u_x, u_y, u_z\}$ is the fluid velocity, E is the total energy, and $\left(c_1, \dots, c_{\mathrm{n}_{sp}}\right)$ are the mass fractions of the n_{sp} gas species in the domain. Neglecting viscous terms for brevity, the governing equations can be summarized as

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{\nabla}\cdot\boldsymbol{F} + (\boldsymbol{\nabla}\cdot\boldsymbol{w})\boldsymbol{U} = \boldsymbol{0}$$

where $\boldsymbol{w}$ is the transport velocity and $\boldsymbol{F}$ is the convective flux, expressed as

$$\boldsymbol{F_x} = (u_x - w_x)\boldsymbol{U} + \begin{Bmatrix} 0 \\ p \\ 0 \\ 0 \\ pu_x \\ 0 \\ \vdots \\ 0 \end{Bmatrix}; \boldsymbol{F_y} = \left(u_y - w_y\right)\boldsymbol{U} + \begin{Bmatrix} 0 \\ 0 \\ p \\ 0 \\ pu_y \\ 0 \\ \vdots \\ 0 \end{Bmatrix} \text{ and } \boldsymbol{F_z} = (u_z - w_z)\boldsymbol{U} + \begin{Bmatrix} 0 \\ 0 \\ 0 \\ p \\ pu_z \\ 0 \\ \vdots \\ 0 \end{Bmatrix}$$

and p is the fluid pressure, calculated from an ideal gas equation of state in this case.
For stability, the semi-discrete conservative equation is written as

$$\frac{\partial \boldsymbol{U}_I}{\partial t} + \sum_J \left(\boldsymbol{F}_{IJ} - \boldsymbol{F}_I\right)\boldsymbol{\nabla}\Psi_J(\boldsymbol{x}_I) + \sum_J \left(\boldsymbol{w}_J - \boldsymbol{w}_I\right)\boldsymbol{\nabla}\Psi_J(\boldsymbol{x}_I)\boldsymbol{U}_I = \boldsymbol{0}$$

where $\boldsymbol{F}_{IJ}$ is the numerical convective flux at the midpoint $\boldsymbol{x}_{IJ}$ between particles I and J, calculated from a Rusanov flux [2,3]

$$\boldsymbol{F}_{IJ} = \frac{1}{2}\left(\boldsymbol{F}_{IJ}^{+} + \boldsymbol{F}_{IJ}^{-}\right) - \frac{1}{2}S_{IJ}\Delta\boldsymbol{U}_{IJ}\cdot\boldsymbol{n}$$

where $\boldsymbol{F}_{IJ}^{+}$ and $\boldsymbol{F}_{IJ}^{-}$ are calculated from the following linear reconstructions capped with a min-mod limiter

$$\begin{cases} \boldsymbol{U}_{IJ}^{+} = \left[\boldsymbol{U}_I + \boldsymbol{\nabla}\boldsymbol{U}_I\cdot\left(\boldsymbol{x}_{IJ} - \boldsymbol{x}_I\right)\right]_{Lim} \\ \boldsymbol{U}_{IJ}^{-} = \left[\boldsymbol{U}_J + \boldsymbol{\nabla}\boldsymbol{U}_J\cdot\left(\boldsymbol{x}_{IJ} - \boldsymbol{x}_J\right)\right]_{Lim} \end{cases},$$

$\boldsymbol{n}$ is the unit vector pointing from point $\boldsymbol{x}_I$ to point $\boldsymbol{x}_\mathrm{J}$, $\Delta\boldsymbol{U}_{IJ} = \boldsymbol{U}_J - \boldsymbol{U}_I$ and $S_{IJ} = max\left(\boldsymbol{u}_{IJ}^{+}\cdot\boldsymbol{n} + c_{IJ}^{+}, \boldsymbol{u}_{IJ}^{-}\cdot\boldsymbol{n} + c_{IJ}^{-}\right)$ where $\boldsymbol{u}_{IJ}^{+}$ and $\boldsymbol{u}_{IJ}^{-}$ are the reconstructed fluid velocities at the midpoint, and c_{IJ}^{+} and c_{IJ}^{-} are the reconstructed soundspeeds at the midpoint.
Explicit derivatives of MLS approximation functions being computationally expensive to calculate, implicit gradients are used instead in this work [4].

© 2024 Copyright by Ansys Inc.

3 Boundary Conditions for Airbag Deployment

In the rest of this work, free-slip boundary conditions are imposed on all surfaces other than inlets and outlets, unless otherwise specified.

3.1 Inflator Orifice as Inlets

A key aspect of airbag deployment simulations is the characterization of the inflator. Industry practice typically involves providing two time-dependent curves for each gas species to represent the gas inflow: a mass-flow rate curve and a total temperature curve. Unlike methods such as CPM, where this data is used to define source points for particle emission, CPG being a CFD solver requires transforming the information into an inlet boundary condition. This necessitates certain assumptions. A set of surface elements is selected to represent the inflator orifice, serving as the inlet boundary for the flow. For accurate simulation results, the surface must be well-represented by the local point cloud in the inflator region. Since inflator orifices are generally very small relative to the overall domain size, the orifice area is artificially expanded by selecting multiple elements around it. Although this ensures that the total mass and energy entering the domain are accurate, fine local flow details are lost. This can be seen as a necessary local smoothing of the flow around the inflator due to the significant size difference between typical inflator orifices and the overall simulation domain.
The next step is to transform the given mass flow rate $\dot{m}$ and total temperature T_0 into suitable boundary conditions. An important assumption we make is that the flow is either subsonic or choked: in other words, the fluid velocity at the inlet is not allowed to exceed the local sound speed. From the mass flow rate and inlet area, the quantity $(\rho v)_{\text{inlet}}$ is fixed as

$$(\rho v)_{\text{inlet}} = \frac{\dot{m}}{A}$$

but the fluid density and velocity are not straight-forward to deduce. From the provided total temperature T_0 and the following equality

$$T_0 = \frac{T_S}{1 + \frac{\gamma - 1}{2} M^2}$$

where T_S is the static temperature, γ is the heat capacity ratio and M is the Mach number, we can obtain the static temperature for two different scenarios:

1. We first assume that the flow is choked, in which case $M = 1$ and this relation yields $T_S = \frac{\gamma+1}{2} T_0$, from which we can deduce the local sound speed $c = \sqrt{\gamma R_s T_S}$ where R_s is the specific gas constant. Using the assumption that the flow is choked, the inlet flow velocity is then prescribed as $v = c$ and the fluid density is imposed at $\rho = \frac{\dot{m}}{vA}$. The fluid pressure is then calculated as $p = \rho R_s T_S$. This pressure is then compared to the interpolated downstream pressure. If the downstream pressure is higher, the choked flow assumption does not hold, and the inlet velocity is therefore subsonic.
2. If the choked assumption is not verified, we set the inlet pressure p equal to the interpolated downstream pressure and use the following equality on enthalpy

$$\int_0^{T_0} C_p(T) dT = \int_0^{T_S} C_p(T) dT + \frac{v_{\text{inlet}}{}^2}{2}$$

with $v_{\text{inlet}} = \frac{(\rho v)_{\text{inlet}} R_s T_S}{p}$ to obtain the static pressure. We can then calculate the fluid density $\rho = \frac{p}{R_s T_S}$ and the fluid velocity $v = \frac{\dot{m}}{\rho A}$.

In either case, the fluid density, velocity and static temperature are imposed, from which the total energy can also be calculated. The mass fraction of each species i is also imposed from the mass flow rate curves: $c_i = \frac{\dot{m}_i}{\sum_{j=0}^{n_{\text{sp}}} \dot{m}_j}$, hence the entire vector of conservative variables $\boldsymbol{U}$ is imposed on the inflator orifice boundary.

3.2 External Vents as Outlets

Vent holes are a key component in airbag modeling. In traditional CFD solvers, an external vent can be viewed as an “outlet” condition where gas is allowed to escape (and sometimes to reenter, depending on the modeling approach). For CPG the implementation works as follows: an external surface area is

© 2024 Copyright by Ansys Inc.

selected as a vent hole. Then, the velocity direction at a given time automatically determines the state of the vent, whether it is in backflow mode or in outflow mode. If in backflow mode, pressure and temperature are imposed, and the gas fraction of air is set to unity (0.0 to all other gases). If in outflow mode, only the pressure is imposed. Whenever pressure or temperature are imposed, P_{atm} and T_{atm} will be used unless overwritten by user defined load curves. One last condition for the vent hole treatment is in the case of supersonic outflow, user-defined pressure values are ignored and replaced by the upstream pressure.

At this stage it is worth noting that outflow conditions are notoriously difficult to model and keep stable in traditional CFD codes. CFD engineers are routinely told to put their outflow conditions “far from the action” where the flow is quiet, and nothing happens i.e. where the assumption of relatively straight flow with uniform pressure holds. In airbag simulations, this is a luxury that is unattainable, and the subject of modeling vent holes is prone to further investigations and improvements with the CPG method. Currently, the recommendation is to try and apply a certain care to the initial surface selection and attempt to ensure that the mesh quality remains reasonable throughout the run (e.g. CPM airbag modelers frequently use materials that have no stiffness and can lead to extreme mesh distortion).

Finally, because CPG is a continuous method solving for the flow field everywhere particles are present, internal vents should simply be modelled as geometric ‘holes’ whether they are present from the start of the simulation or only activating after a specific criterion has been reached (usually based on the flow pressure).

3.3 Treatment of Fabric Porosity

Porosity is another important component of airbag modeling. Gases can leak through external or internal fabric material, sometimes significantly affecting the deployment process. Currently, porosity is supported for external surfaces. When a fabric material is detected on an external surface (see ***MAT_FABRIC**), the CPG solver will calculate the pressure difference between the internal gas pressure and P_{atm} and use the user provided load curve ($v = f(dP)$) to impose the velocity in the normal direction (the sign of the pressure difference will determine the velocity direction i.e. whether gas is leaking out or outside air is coming in).

3.4 Heat Exchange and Thermal Coupling

It is possible to model the thermal exchange with external surfaces by assigning a heat transfer coefficient (see **HCONV** in ***AIRBAG_CPG**). The CPG solver will add a heat source or sink term on wall particles based on the choice of heat transfer coefficient, local surface area, the current local temperature and atmospheric temperature T_{atm}.

While still under development, it is also worth mentioning that coupling between CPG and LS-DYNA’s implicit thermal solver is available for shell surfaces. When the thermal solver is present, the temperature at the surfaces will be applied as boundary conditions for CPG wall particles while the CPG solver will return a heat flux term based on the aforementioned input.

4 CPG Keywords

The main keyword, ***AIRBAG_CPG**, was designed to be as identical as possible to its CPM counterpart ***AIRBAG_PARTICLE**, see Table 1. A first part set **sid1** is defined to identify the components that will interact with the gas, and a second part set **sid2** lists the internal components. The list of parts contained in **sid1** but not in **sid2** should correspond to the manifold surface of the airbag domain.

Fig. 1 offers a snapshot of the ***AIRBAG_CPG** keyword. Readers are referred to the latest LS-DYNA Keyword User’s Manual for more details on each field in this keyword. For the benefit of users already familiar with the ***AIRBAG_PARTICLE** keyword, the main differences are highlighted below:

- ~~**np**~~ → **hlen:** The CPG solver is automatically adding and removing particles to properly sample the domain as it deforms. The average interparticle distance is controlled by the **hlen** field, and replaces the **np** field in CPM which controls the number of particles.
- **nvent:** In CPG, only external vents should be specified. Internal vents are automatically handled by the solver and don’t need to be meshed or specified in the airbag part sets. See section 3.2 for more information on how vents are treated in CPG.

© 2024 Copyright by Ansys Inc.

- ~~**iair**~~ → **[ ]:** CPG is a CFD solver, and therefore the initial air inside the airbag is always modeled. As a result, the iair field becomes redundant, and a behavior similar to **iair=2/4** is achieved.
- **npair** → **[ ]:** CPG particles essentially form a point cloud through which the gas travels. There is no notion of air particles vs inflator particles: Each gas species is free to advect through the point cloud. Consequently, the **npair** field becomes obsolete.
- ~~**nid1**~~ → **ssh1:** Orifices are defined as shells (positive value) or shell sets (negative values) in CPG, as opposed to nodes. The orifice area is automatically calculated based on the area of the designated shells.

***AIRBAG_CPG**							
sid1	stype1	sid2	stype2	[ ]	npdata	[ ]	[ ]
hlen	unit	[ ]	Tatm	Patm	**nvent**	[ ]	[ ]
[ ]	ngas	norif	nid1	nid2	nid3	[ ]	[ ]
pair	tair	xmair	Aair	Bair	cair	**[ ]**	[ ]
lcm1	lct1	xm1	a1	b1	c1	infg1	[ ]
ssh1	[ ]						

*Table 1: Main syntax of ***AIRBAG_CPG** keyword. Differences with ***AIRBAG_PARTICLES** are emphasized in bold.*

Other related keywords include

- ***DEFINE_CPG_GAS_PROPERTIES** can be used to specify up to fourth order temperature dependent heat capacity at constant pressure ($C_P(T)$), fluid viscosity and thermal conductivity.
- ***INITIAL_CPG** in conjunction with ***DEFINE_CPG_REGION** is used to initialize different gas properties in different regions of the domain.
- ***MESH_SIZE_SHAPE** can be used to specify different interparticle distances **hlen** in different regions of the domain. This is particularly useful to locally refine the point cloud around the inflator, and smoothly transition to a larger interparticle distance away from the inflator. Multiple ***MESH_SIZE_SHAPE** keywords can be defined, and each region can follow a structural node. The interparticle distance can be specified in a ***DEFINE_FUNCTION**, allowing for a smooth ramping between different **hlen** values.

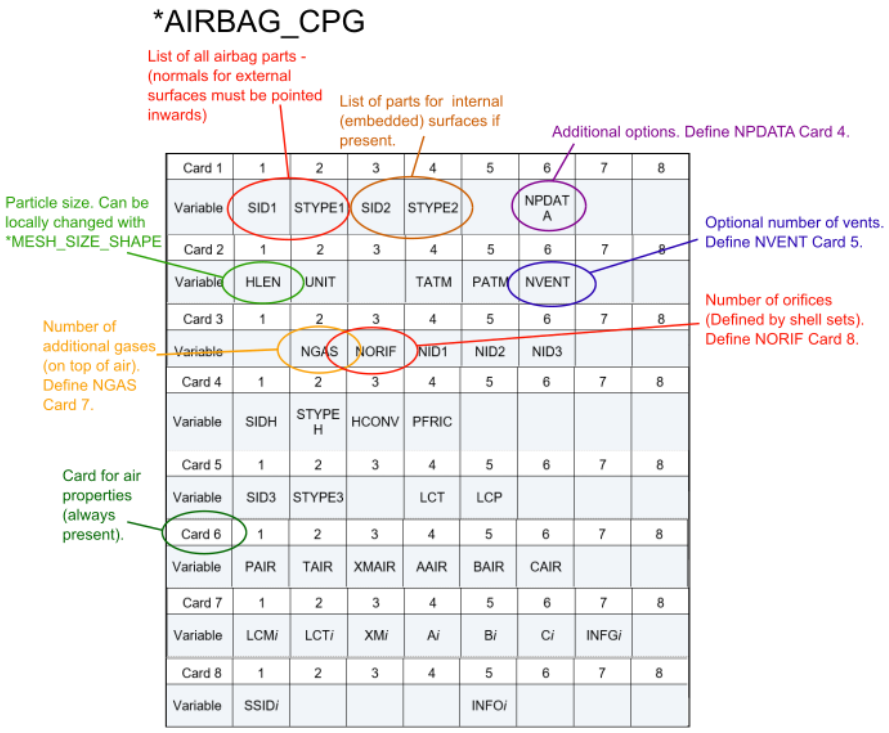

*Fig. 1:The current ***AIRBAG_CPG** keyword structure with the main fields highlighted.*

© 2024 Copyright by Ansys Inc.

5 Verification Cases

5.1 Sod Shock Tube

The Sod shock tube problem is one of the most common benchmarks for compressible fluid solvers [5]. A tube is separated in two sections of equal length with different initial conditions, as listed in Table 2. The profiles of fluid velocity, density and pressure at $t = 0.2$ are given in Fig. 2- Fig. 4 and show very good agreement between the CPG results and analytical solutions. Fig 5 shows the time evolution of the pressure field in the domain. The lack of spurious oscillations at the points of discontinuity also illustrates that the use of the Rusanov numerical flux with min-mod limiter is appropriate in this scenario.

Total length of tube	1.0
Left side density ρ_L	1.0
Left side pressure p_L	1.0
Right side density ρ_R	0.125
Right side pressure p_R	0.1

Table 2: Gas properties on the left and right side of the shock tube.

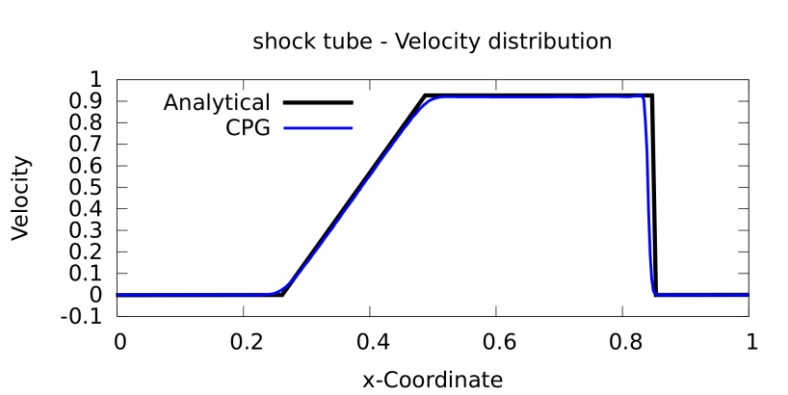

Fig. 2: Velocity distribution along the tube length at t=0.2, analytical vs. CPG.

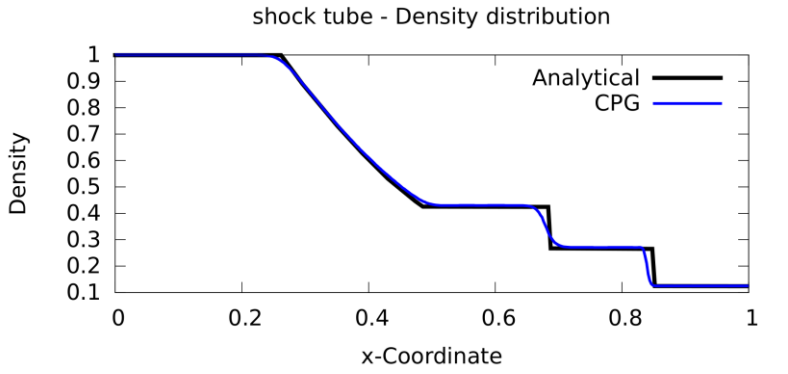

Fig. 3: Density distribution along the tube length at t=0.2, analytical vs. CPG.

© 2024 Copyright by Ansys Inc.

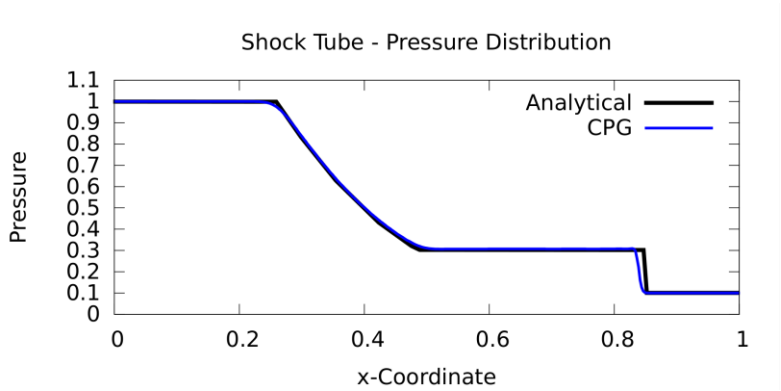

Fig. 4: Pressure distribution along the tube length at t=0.2, analytical vs. CPG.

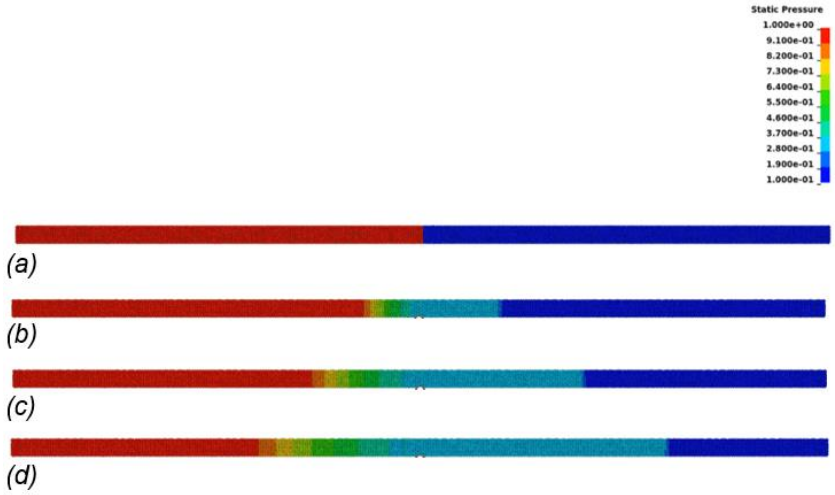

Fig 5: Time evolution of the pressure distribution along the tube length.

5.2 Poiseuille Flow

Another typical benchmark problem is the study of an isothermal compressible flow through a cylindrical pipe, often referred to as a Hagen-Poiseuille flow [6]. This is the only case where we use no-slip boundary conditions on the walls of the cylinder and serves as a verification test for the implementation of the viscous flow that arises from on a non-zero laminar viscosity. The choice of viscosity value is voluntarily exaggerated for the purposes of this test and would not be realistic in any “real” airbag deployment configuration.

A 50mm long, 5mm diameter pipe has an inlet pressure of 1.2 bar and outlet pressure of 1.0 bar. The volumetric flow rate at the outlet is monitored until the expected parabolic velocity profile at the outlet develops (Fig. 6) and the volumetric flow rate reaches a steady-state condition, at which point the analysis is stopped. The parameters for this test, analytical volumetric flow rate at the outlet and numerical result once steady-state conditions are obtained are listed in Table 3. The very good agreement between theoretical and numerical values serves as a good verification case.

Dynamic Viscosity	0.2 Pa.s
Tube Length | Tube Diameter	50mm | 5mm
Inlet Pressure | Outlet Pressure	1.2 bar | 1.0 bar
Analytical Volumetric Flow Rate	5.4e-4 m3/s
Numerical Flow Rate, h = 0.3mm	5.5e-4 m3/s

Table 3: Parameters and results for the isothermal compressible Hagen-Poiseuille flow.

© 2024 Copyright by Ansys Inc.

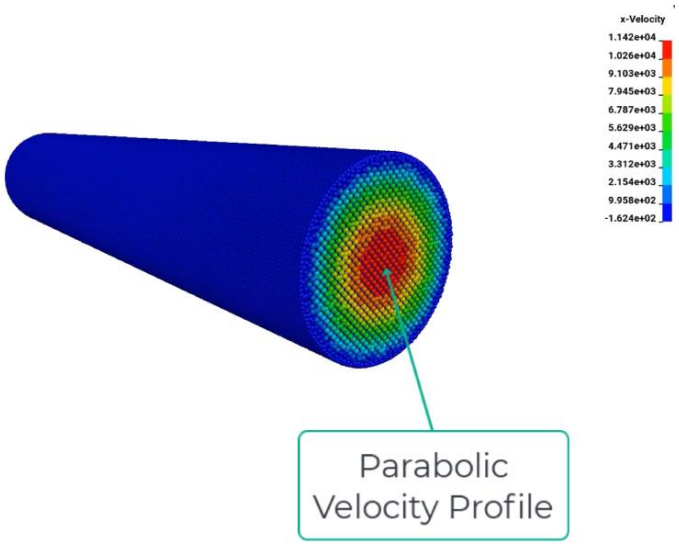

Fig. 6: Poiseuille flow example using CPG.

5.3 Tank Test

An airbag inflator is fired inside a closed tank (Fig. 7), and the pressure is measured on the tank side walls and top. The total temperature of the gas at the inlet is fixed at $T_0 = 400K$, at the prescribed mass flow rate is given in Fig. 8. The point cloud is locally refined around the inflator for better accuracy. This kind of simulation is also a good test to check whether the solver can capture the correct inflow of mass and energy at the inlet. If the inflator surface area is not well represented by the local point cloud discretization, the numerical pressure will start to deviate from the analytical pressure. In this case, Fig. 9 illustrates that this boundary condition is properly represented, and the expected pressure is reached.

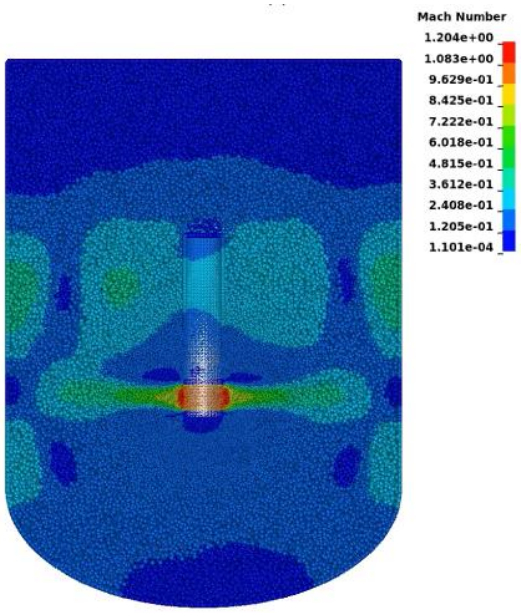

Fig. 7: Cross section of the tank test at t=0.125s, colored by Mach number.

© 2024 Copyright by Ansys Inc.

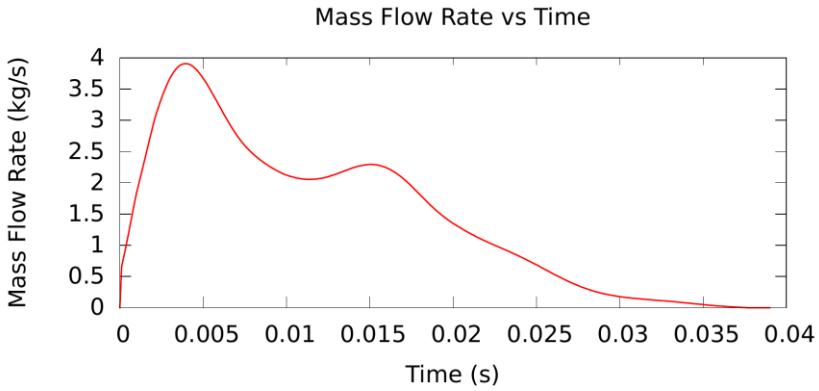

Fig. 8:Prescribed mass flow rate at the inflator orifice for the tank test study.

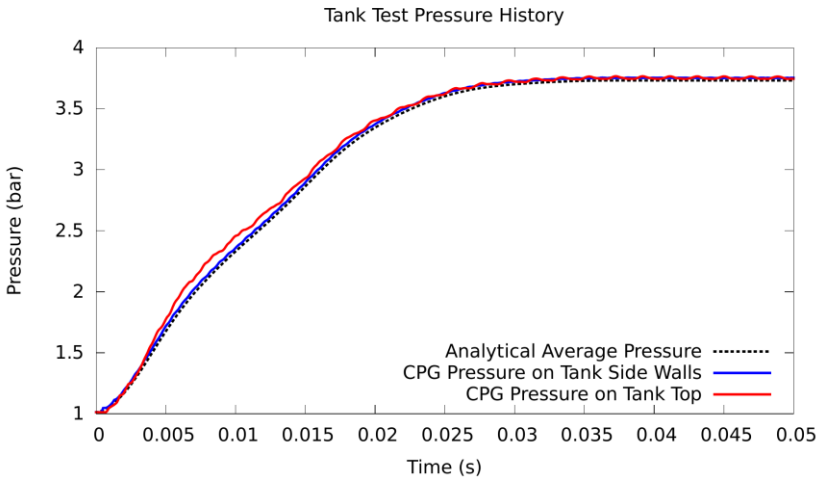

Fig. 9: Pressure history on the tank side walls and top, compared with the analytical average pressure.

6 Validation with Curtain Airbag

The final test involves a curtain airbag. The LS-DYNA model as well as the experimental footage were provided by JSOL Corporation. The CPG results are compared with experimental footage in Fig. 10-Fig. 13 and show very good agreement during the deployment phase. A more detailed view of the local features of the flow around the inflator orifice is provided in Fig. 14, illustrating the capability to infer the value of local quantities, which can be difficult to achieve with CPM.

© 2024 Copyright by Ansys Inc.

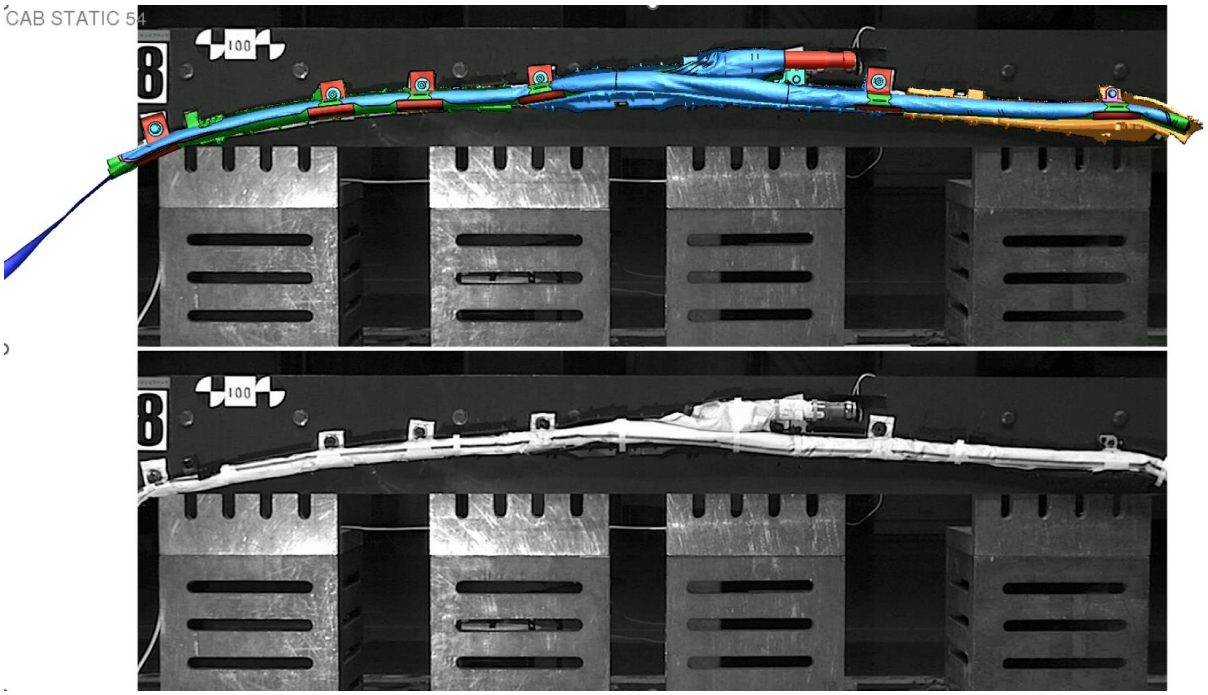

Fig. 10: Demonstration curtain airbag at t=0ms. CPG result overlaid on experimental footage (top), experimental footage (bottom). Model and footage courtesy of JSOL Corporation.

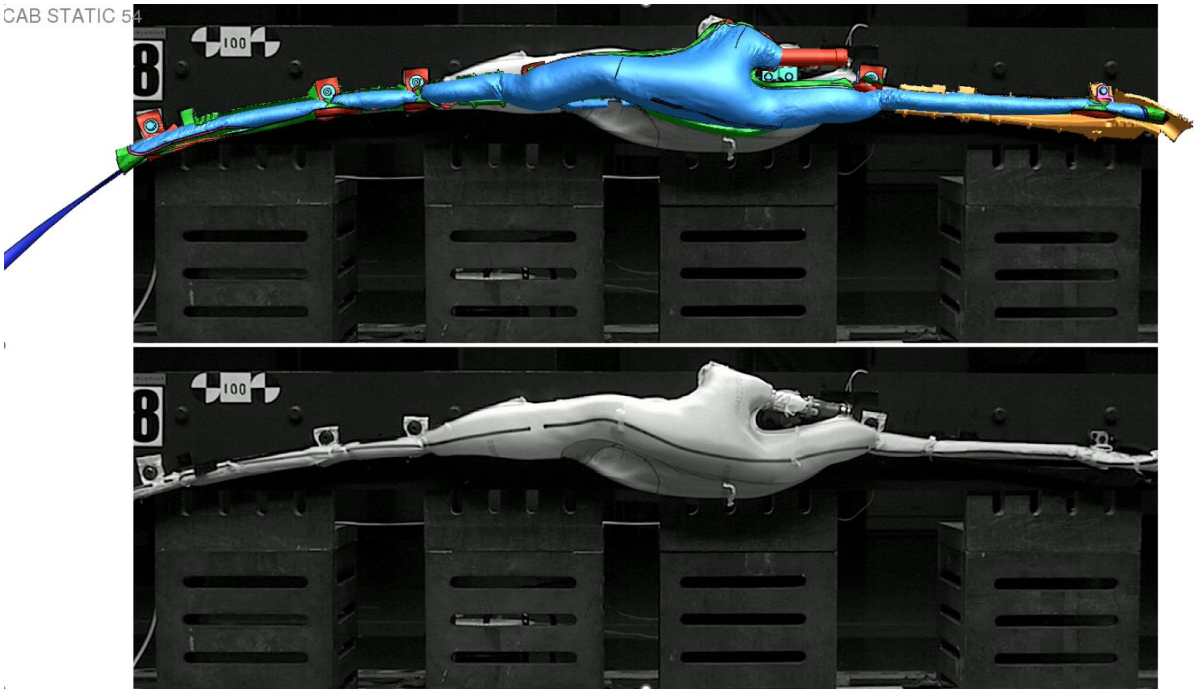

Fig. 11: Demonstration curtain airbag at t=5ms. CPG result overlaid on experimental footage (top), experimental footage (bottom). Model and footage courtesy of JSOL Corporation.

© 2024 Copyright by Ansys Inc.

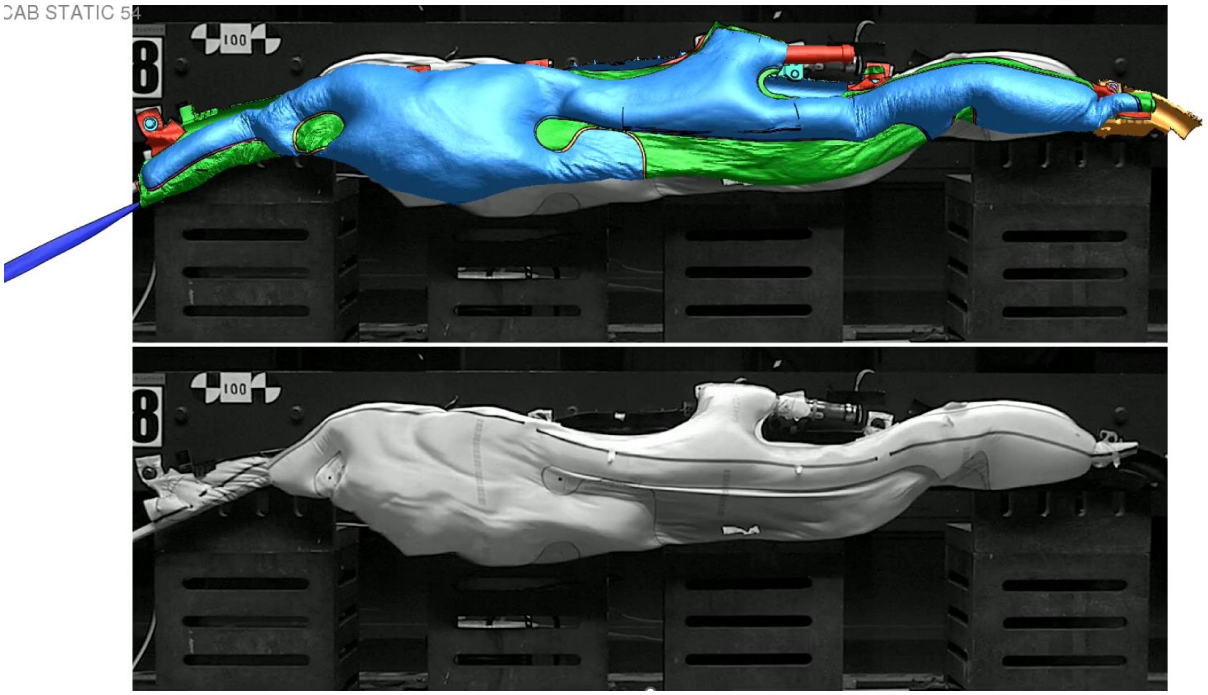

Fig. 12: Demonstration curtain airbag at t=10ms. CPG result overlaid on experimental footage (top), experimental footage (bottom). Model and footage courtesy of JSOL Corporation.

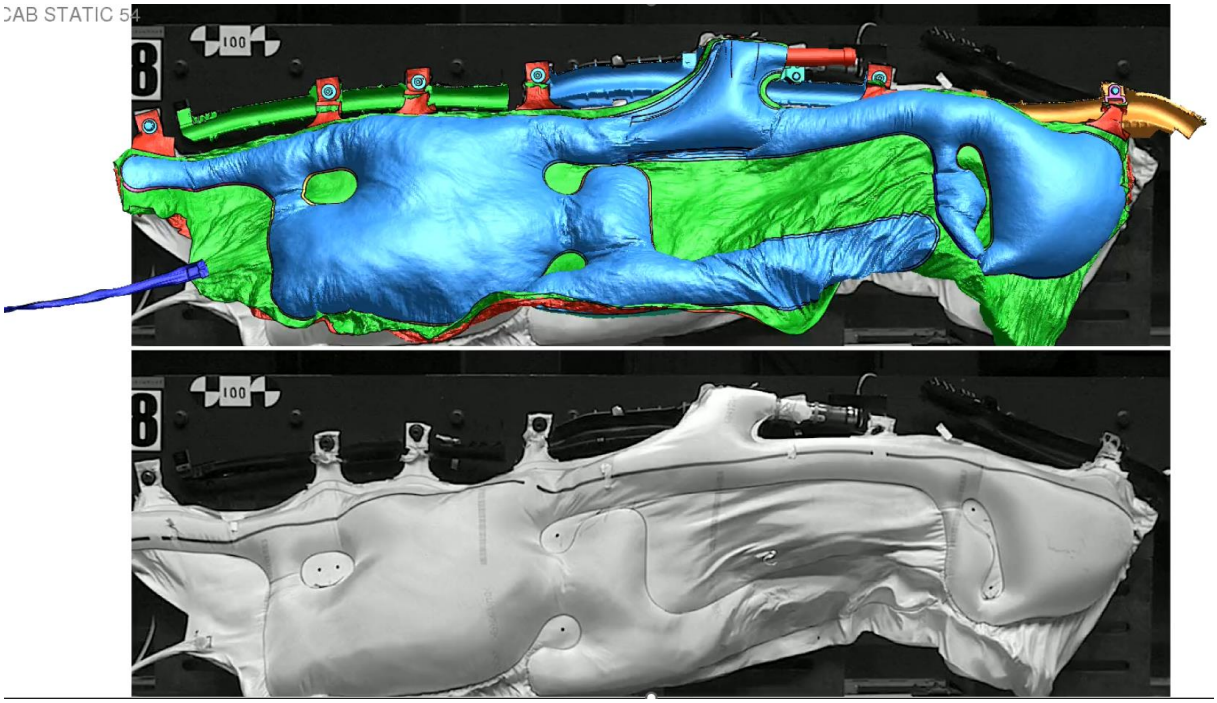

Fig. 13: Demonstration curtain airbag at t=20ms. CPG result overlaid on experimental footage (top), experimental footage (bottom). Model and footage courtesy of JSOL Corporation.

© 2024 Copyright by Ansys Inc.

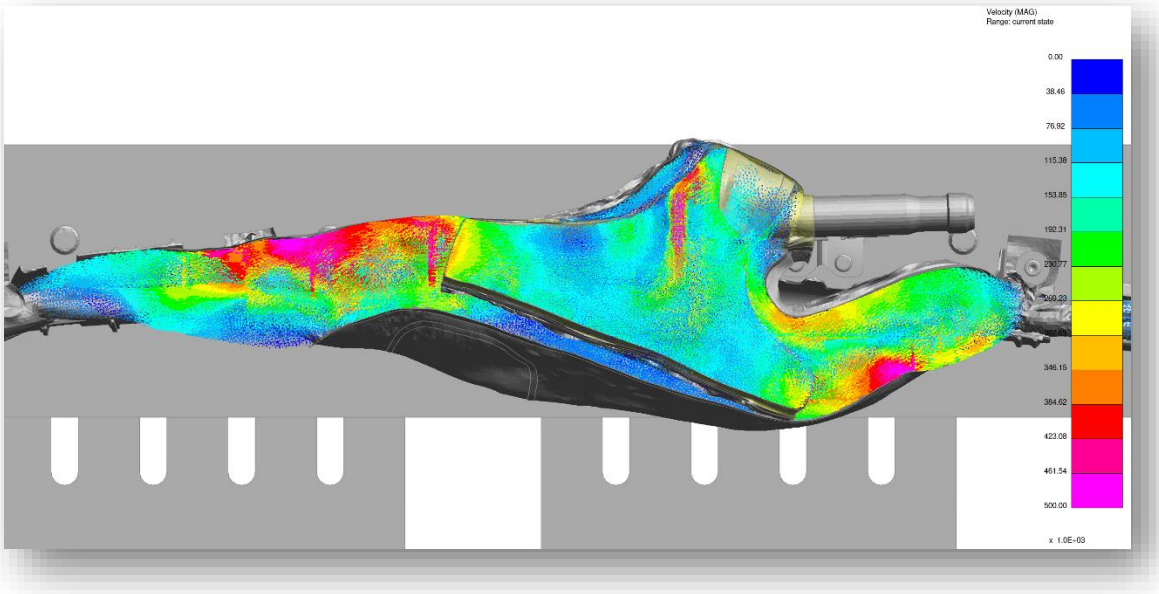

Fig. 14: Gas velocity around the airbag inflator. Particles are colored by velocity magnitude, and vectors show the flow direction.

7 Current Status and Future Work

CPG is released in R16. Its main objective, as described in this paper, is to provide users with all the basic tools necessary to conduct their own airbag analysis and investigations.

Future developments will focus on two key areas: Providing new necessary features (or improve on existing ones) and improving computational efficiency. For feature enhancement, priority will be given to a more flexible treatment of vent surfaces, minimizing the need for users to closely monitor mesh distortion in these regions. Further validation of the porosity implementation is needed, and we plan to investigate support for internal porous media fabric as well as explore heat exchange processes on internal surfaces. To improve calculation speed, we will explore alternative algorithms for particle neighbor downselection near surfaces and enhance MPP load distribution. Adding the ability to coarsen the point cloud in selected region could also significantly improve overall performance. Other developments are being considered and will be prioritized based on user feedback.

Finally, we would like to present the development of this solver as an opportunity to deepen our collective understanding of airbag deployments. Given the complexities of the physics involved, success will rely on close collaboration between experimental testing and software development supported by extensive feedback and open exchanges. This virtuous cycle is already underway, as demonstrated by the results and experimental comparisons provided in this paper, courtesy of JSOL Corporation. We welcome further discussions to drive future developments. We would also like to express our gratitude to Toyoda Gosei Co., Ltd. for their continued contributions to the development of CPG by providing numerous physical tests specifically designed to validate this new solver. Finally, we greatly appreciate the invaluable contributions of Richard Taylor (Arup), which have been instrumental to our progress.

© 2024 Copyright by Ansys Inc.

8 Literature

[1] Lancaster, P. and Salkauskas, K., 1981. Surfaces generated by moving least squares methods. Mathematics of computation, 37(155), pp.141-158.
[2] VV Rusanov, On Difference Schemes Of Third Order Accuracy For Nonlinear Hyperbolic (Systems Journal of Computational Physics Vol. 5) pp. 507-516
[3] L. Ramírez, A. Eirís, I. Couceiro, J. París, X. Nogueira, An arbitrary Lagrangian-Eulerian SPH-MLS method for the computation of compressible viscous flows, Journal of Computational Physics
[4] S. Li, W.K. Liu, Reproducing kernel hierarchical partition of unity Part I: Formulation and theory, Internat. J. Numer. Methods Engrg. 45 (1999) 251–288
[5] Sod, G.A., 1978. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. Journal of computational physics, 27(1), pp.1-31.
[6] Chassaing, P., 1982. Mecanique des fluides. Elements d'un premier parcours. Cepadues Editions

© 2024 Copyright by Ansys Inc.