

USING THE DIRECT SIMULATION MONTE CARLO APPROACH FOR THE BLAST-IMPACT PROBLEM

Anupam Sharma*, Lyle N. Long† and Ted Krauthammer‡
The Pennsylvania State University
University Park, PA 16802, U.S.A.

The viability of using the Direct Simulation Monte Carlo (DSMC) approach to study the blast-impact problem is assessed. An uncoupled analysis of the problem is performed where the solid body is not allowed to move or deform. The numerical simulations are designed to mimic the actual shock-tube experiments. Our code is validated against the shock-tube (Riemann) problem. A novel approach to model the inflow boundary condition is presented which can be used with both particle and continuum methods. A generic implementation of the solid boundary condition for particle methods is described which can easily and efficiently handle arbitrary-shaped bodies. This approach is demonstrated by computing load definition for two model geometries - a box and an 'I' shaped beam.

Introduction

TRAGIC mishaps such as the attacks on the World Trade Center and the USS Cole have necessitated the design and construction of blast-resistant structures that can be used to protect critical buildings. This is an extremely challenging task since it is so difficult to estimate the magnitude of the blast, and the location (inside/outside a building) of the explosive. Since we cannot estimate the "threat" we are fighting against, we have to design our structures for "performance", the performance defined as the ability to withstand unexpected forces. A preliminary requirement in the design of such blast-resistant structures is to have a time history of the pressure on the body which is subjected to a blast-impact. This is called load definition. This study is aimed at obtaining accurate load definition on the body in a reasonable amount of time.

After an explosion, a blast (shock) wave is formed which travels outward in three dimensions. The intensity of the wave continuously decreases with the distance from the explosion center. A typical pressure history from an explosion at a point some distance off the explosion center is shown in Fig. 1

Bleakney *et.al.*² were the first to experimentally investigate the loading on structures due to impact from a shock wave. Although the pressure falls steadily behind the shock front, the rate of decay is so slow that the first few hundred feet of the wave can be considered flat topped.¹ Hence, the initial loading on the structures may be studied using shock-tube experiments (c.f. ref.²). Bleakney *et.al.*² conducted a variety of shock-tube experiments to provide a large amount of basic data on blast-loading for future analyses by others. However, unfortunately, no significant follow-up investigation of this problem has been published.

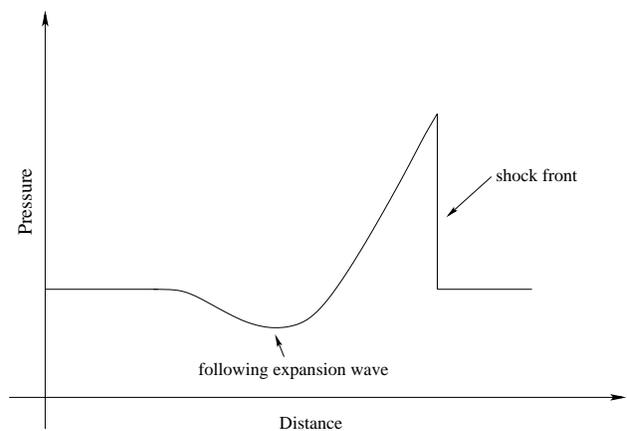


Fig. 1 Pressure/distance variation a few seconds after the explosion of a hydrogen bomb.¹

Although shock-tube experiments are not too expensive to perform, scientists are now becoming more and more inclined towards doing numerical experiments. A numerical experiment is a simulation of the actual experiment using a computer. This inclination is partly because of the rapid growth in computer speed and partly because of the ease in performing these numerical experiments. In fact, the computer simulations in most cases are cheaper than the actual experiments. The authors are motivated by the same reasons to use a computer as a tool to solve the blast-impact problem. The solution to this problem requires a coupling between fluid dynamics and structural dynamics. In this paper, we neglect the coupling and assume that the solid body does not deform or move because of the impact. Hence, we are solving a simplified problem which lies completely in the fluid dynamics domain. Our ultimate goal, however, is to develop a simulation approach that includes deformable structures.

A hierarchy of mathematical models are available to solve fluid dynamics problems. These models have varying degrees of approximation and can be broadly

*Ph.D. Candidate, Aersp. Engr., axs455@psu.edu

†Prof., Aersp. Engr., Assoc Fellow, AIAA. lnl@psu.edu

‡Prof., Civil Engr. tedk@psu.edu

classified into two groups - 1) continuum methods which treat the fluid as a continuous medium and, 2) particle methods which treat the fluid as made up of particles. Although the continuum approach is popular for many engineering problems, it has a limited range of applicability. The continuum approximation fails for rarefied gas flows, and flows in nanodevices. On the other hand, particle methods do not make the continuum approximation, are more intuitive, easy to apply, and can give approximate results (with statistical scatter) much faster than the continuum methods.

The different particle methods in use are the molecular dynamics model, Monte Carlo methods and the direct solution of Boltzmann equation.³ The most fundamental is the ‘molecular dynamics’ model. Here the particles are moved according to the Newton’s law ($F = ma$). It can even account for quantum mechanical effects. However, it is extremely computationally intensive and is not extensively used for engineering problems.

Both Monte Carlo approaches and the Boltzmann equation are derived from the Liouville equation. The continuum equations (e.g. Navier-Stokes) can be derived from the Boltzmann equation if one assumes the Knudsen number is small. The term “Monte Carlo” was adopted by von Neumann for methods involving statistical techniques, such as the use of random numbers, to find the solutions to mathematical or physical problems. The first documented application of a Monte Carlo method appears in a paper by G. Comte de Buffon in 1777. He described an experiment in which a needle of length L is thrown randomly onto a floor ruled with parallel straight lines a distance D apart. He then estimated the probability that the needle would intersect a line, by throwing the needle many times and calculating the ratio of the number of throws intersecting a line to the total number of throws. This is referred to as “Buffon’s Needle Experiment”. Later, Laplace suggested that the same idea could be used to evaluate π from the throws of the needle. Lord Kelvin⁴ used the Monte Carlo method in 1901 to perform time integrals which appeared in his kinetic theory of gases. However, it was only in 1940s that the Monte Carlo methods were developed enough to be used for solving engineering problems.

The Direct Simulation Monte Carlo (DSMC) method has gained a lot of popularity since it was developed by Bird.^{5,6} DSMC has been successfully used for hypersonic flows^{5,7,8} where the continuum approximation breaks down and the traditional Navier-Stokes equations are not valid. In fact, DSMC has become *de facto* the principal tool to investigate such flows. DSMC is also very useful for modeling flows involving chemical reactions.^{9,10} The DSMC approach is well suited for our problem because of the ease of modeling the solid body. An arbitrary shaped body can be handled with the same ease as a regular shaped object. A

complicated geometry is extremely difficult to incorporate in conventional computational fluid dynamics (CFD) methods. A new approach by Morris *et.al.*¹¹ of representing the solid body by a dense fluid may be used to model arbitrary shaped bodies, but this is still under development. Also, Long¹² developed a technique for handling very complex bodies in CFD.

In this paper we explore using DSMC to model the blast-impact problem. We developed a Fortran 90 code to solve this problem. Our code is validated against the shock-tube (Riemann) problem. The solid boundary condition is applied in a numerically efficient manner. A novel approach to model the inflow boundary condition is presented which significantly reduces the computation cost. We present the impact study on two two-dimensional solid bodies - a cube and an I-beam for different shock strengths. For the present study, these bodies are not deformed or moved by the shock impact. A deformation in the body relieves the pressure on it, hence we are overestimating the load on the body. Although it is unphysical, it is a good first-hand approximation and it allows for a safer design.

DSMC

The Monte Carlo methods can be used not only for naturally stochastic problems, but also for deterministic problems. The DSMC approach is partly probabilistic and partly deterministic. It is derived from the kinetic theory of gases and it employs the basics of Monte Carlo methods.

The Knudsen number for a flow is defined as the ratio of the mean free path of the gas to the length scale in the problem.

$$Kn = \lambda/L \quad (1)$$

Local length scales may be used in problems where the Knudsen number may change across the domain, for example, $L = \rho/|\nabla\rho|$. Inviscid flows have $Kn = 0$. For $Kn > 0.1$ the approximation for the shear and heat fluxes breaks down and the governing equations do not form a closed set. Hence the Navier-Stokes equations are only valid for very small Knudsen numbers. The DSMC method works for the whole range of Kn provided the fluid is dilute. However, for low Kn , DSMC can be very computationally expensive. Here we present an approach to make it efficient at low Kn . This is not the first attempt to apply DSMC for low Kn flows. DSMC has been shown to work in the near continuum regime by Merkle *et. al.*¹³

DSMC uses a representative set of particles which are concurrently followed through representative collisions and boundary interactions in simulated physical space. Each simulated particle may represent millions of actual gas molecules. The simulated space is divided into a number of cells (grid) to calculate

the macroscopic quantities. The macroscopic quantities are obtained by sampling the molecular (particle) properties over the particles in each cell. The important assumption in the DSMC procedure is that the molecular motion and the intermolecular collisions are uncoupled over the time step. This limits the size of the time step to be of the order of the mean collision time, and the cell size to be of the order of the mean free path. This is, however, a very stringent limit and can be relaxed at a very minimal cost when dealing with continuum flows. Since the boundary layer on the body and the structure of the shock have no significant effect on the load definition, we can ignore these for our computations. Hence, we essentially perform Euler calculations using DSMC with a time step orders of magnitude larger than the mean collision time. According to the kinetic theory, a large number of intermolecular collisions should be performed when the time step is chosen to be very large. This is what makes DSMC computationally expensive for continuum flows. However, for our case, we restrict the number of collisions to a number which is sufficient to bring the flow to thermal equilibrium. Any additional collisions do not improve the accuracy of the computations. The only region which may get adversely affected by this limit on the number of collision is the shock-wave region, but again, we are not interested in the details of the shock structure. This approach works extremely well as is shown in the following section.

Code Validation

We validate our code against the shock-tube (Riemann) problem. The shock-tube is initially divided into two chambers separated by a diaphragm. One chamber contains a hot gas at high pressure and density and the other contains a cold gas at low pressure and density. The gases in the two chambers may be different but in our simulations they are the same. The diaphragm is then burst and the hot gas (driver) is allowed to expand into the cold (driven) section. A shock wave and an expansion wave are formed which travel in opposite directions. The shock wave moves at supersonic speed into the driven section and the expansion wave travels into the driver section. Although the expansion wave travels into the driver section, the motion of the gas is always in the direction of the shock wave. A schematic of the shock-tube problem with the pressure distribution both before and after the diaphragm is burst is sketched in Fig. 2. There are four distinct regions marked ‘1’, ‘2’, ‘3’ and ‘4’ in Fig. 2. Region ‘1’ is the cold gas which is undisturbed by the shock wave. Region ‘2’ contains the gas immediately behind the shock traveling at a constant speed. The ‘contact surface’ across which the density and temperature are discontinuous lies in this region. The region between the head and the tail of the expansion fan is

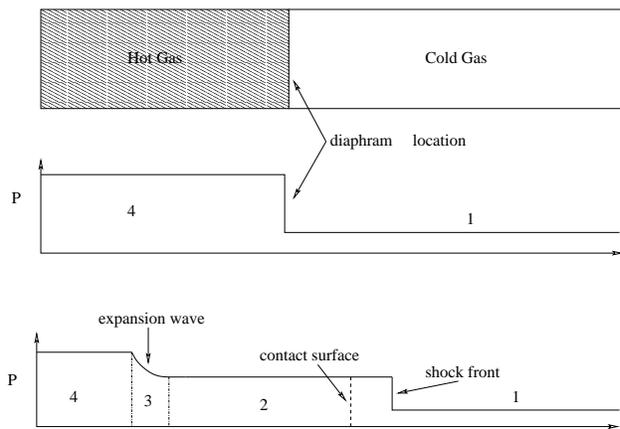


Fig. 2 A schematic of a shock-tube experiment; the pressure distribution at $t = 0$ and some time after the diaphragm is burst.

mark region ‘3’. In ‘3’ the flow properties change gradually since the expansion process is isentropic. Region ‘4’ denotes the undisturbed hot gas. The interested reader may refer to the monologue on shock tubes by Wright¹ for further reading.

The analytical solution to the Riemann problem is available (c.f. ref.¹⁴). We assume that the ratio of the specific heat constants of the gas, γ does not change with temperature (which is valid for the low temperature range). Our simplified equations (Eqs. 2) in region ‘2’ are obtained using the normal shock relations.

$$\begin{aligned}
 \frac{p_4}{p_1} &= \frac{p_2}{p_1} \left\{ 1 - \frac{(\gamma - 1)(a_1/a_4)(p_2/p_1 - 1)}{\sqrt{2\gamma} [2\gamma + (\gamma + 1)(p_2/p_1 - 1)]} \right\}^{\frac{-2\gamma}{\gamma - 1}} \\
 u_2 &= \frac{a_1}{\gamma} \left(\frac{p_2}{p_1} - 1 \right) \left(\frac{2\gamma/(\gamma + 1)}{p_2/p_1 + (\gamma - 1)/(\gamma + 1)} \right)^{1/2} \\
 \frac{T_2}{T_1} &= \frac{p_2}{p_1} \left(\frac{(\gamma + 1)/(\gamma - 1) + p_2/p_1}{1 + (p_2/p_1)(\gamma + 1)/(\gamma - 1)} \right) \\
 p_2 &= \rho RT_2
 \end{aligned} \tag{2}$$

The velocity of the gas in region ‘2’ is constant throughout and is equal to u_2 . The method of characteristics yields the solution in region ‘3’ ($-a_4 \leq x/t \leq u_3 - a_3$).

$$\begin{aligned}
 u_3 &= \frac{2}{\gamma + 1} \left(a_4 + \frac{x}{t} \right) \\
 \frac{p_3}{p_4} &= \left[1 - \frac{\gamma - 1}{2} (u_3/a_4) \right]^{2/(\gamma - 1)} \\
 \frac{p_3}{p_4} &= (\rho_3/\rho_4)^\gamma = (T_3/T_4)^{\gamma/\gamma - 1}
 \end{aligned} \tag{3}$$

We used the same setup as in an experiment for our numerical calculations. The flow properties are averaged in the ‘y’ direction as this is a one dimensional problem. A sample comparison is provided in Fig. 3 for the following set of values in SI units:

$$\rho_1 = 1.226; \rho_4 = 4.226; T_1 = 300; T_4 = 900$$

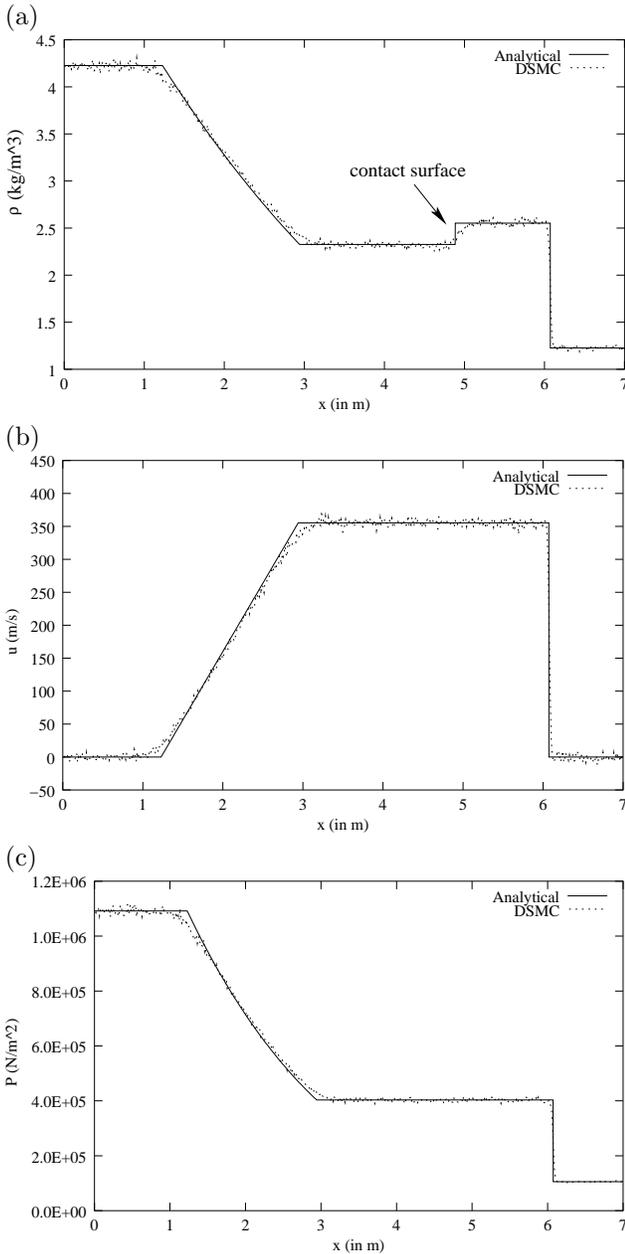


Fig. 3 Code validation against the shock-tube problem at $t=3.6$ ms. (a) density, (b) velocity, and (c) pressure

The DSMC calculations capture the sharp gradients and give an excellent overall match with the analytical solution. The DSMC results show a slight smoothing at the contact surface (refer Fig. 3 (a)) but it is not of critical importance for the problem of interest.

Inflow Boundary Condition in DSMC

In the simulation of a shock tube we do additional work to compute the expansion wave. Since we need to study the impact only with the shock wave, the computation of the expansion wave is unnecessary and should be avoided. One way to achieve this is by applying an appropriate inflow boundary condition. A good choice of the boundary condition is to use the properties in

region ‘2’. These properties do not change with time for an infinitely long tube. Thus we have a time independent inflow boundary condition that can simulate the shock wave of the shock-tube without worrying about the expansion wave. This boundary condition can be used with conventional CFD schemes as well as with particle methods such as DSMC.

The treatment of the inflow boundary condition in DSMC is different than in the CFD schemes. This is because the boundary condition is known in terms of flow variables whereas we need the number, position and velocity of the entering particles. The number flux of the particles, \dot{N}_i can be obtained by integrating the equilibrium distribution function over the entire range of velocity.⁷

$$\dot{N}_i = \frac{n\beta^3}{\pi^{3/2}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-c_0 \cos\theta}^{\infty} (u' + c_0 \cos\theta) \times \exp\{-\beta^2(u'^2 + v'^2 + w'^2)\} du' dv' dw' \quad (4)$$

Using standard integrals and realizing that $\theta = 0$ for the inflow boundary in our case, Eq. 4 can be reduced to

$$\dot{N}_i = \frac{n}{2\pi^{1/2}\beta} \left[\exp(-s^2) + \pi^{1/2} s \{1 + \text{erf}(s)\} \right] \quad (5)$$

where $s = c_0\beta = c_0/c'_m$ is the molecular speed ratio.

The velocities of the entering particles are given by a Maxwellian distribution about the inflow velocity. The velocity of the particles should be such that in the next time step they move into the domain. The entering particles are uniformly distributed on the inflow boundary and moved in by a distance equal to its velocity times the time step. The particles which move out of the domain through the inflow boundary are ignored.

This approach gives excellent results and a comparison with the analytical solution for the shock-tube problem is provided in Fig. 4. We use inflow boundary condition on the left ‘x’ boundary, wall boundary condition on the right ‘x’ boundary and in ‘y’ direction and periodic boundary condition in ‘z’ direction. It should be noted that the inflow boundary cannot be placed arbitrarily close to the body as the reflected shock will change the conditions on the boundary.

Solid Boundary Condition in DSMC

The ultimate goal of a blast-impact simulation is to couple the shock-wave simulation with a structural dynamics model. This would allow the deformation of the solid body in real time as the stress on the body exceeds the yield stress. Although we do not allow deformation of the solid body in the present simulations, a very general approach to model the solid boundary is developed. This will allow us to use the same code with little change when the body is simultaneously deformed.

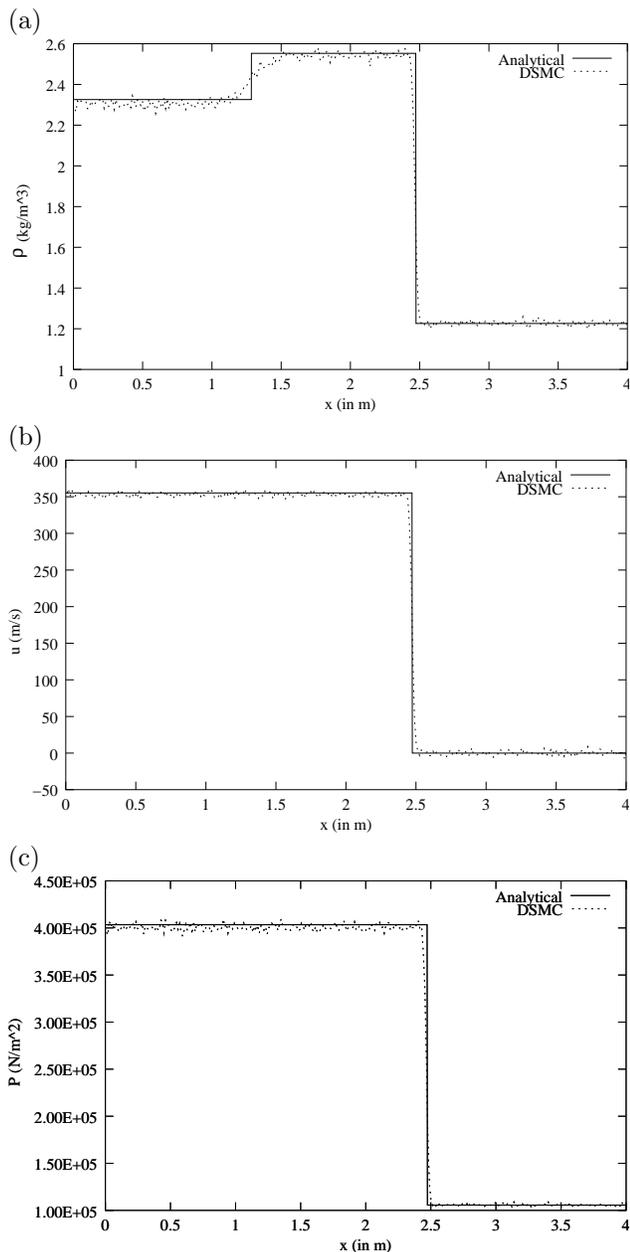


Fig. 4 Verification of the inflow boundary condition. Comparison with the analytical solution of the shock-tube problem at $t=3.6$ ms. (a) density, (b) velocity, and (c) pressure

We implement the inviscid boundary condition in our DSMC calculation by imposing specular reflection of the molecules when they hit a solid surface. The solid surfaces are made of triangular patches to incorporate arbitrary-shaped bodies. The patches may be arbitrarily oriented in 3-dimensions and the molecules are also moving in a 3-dimensional space. We need to find the post-collision position and velocity of the molecules reflected off the solid surface. The probability of a collision of a molecule with a given triangle is very small and therefore it is imperative to devise a computationally efficient algorithm to reject the molecules which do not collide with the triangle.

A quick way to eliminate the molecules which will definitely not collide with the solid body is by checking if the molecule crosses into a box bounding the solid body. It is much cheaper to check the intersection with a bounding box (a cuboid) than an arbitrary solid body since it only requires six logical statements in a subroutine to compare the three Cartesian coordinates (position) with the dimensions of the box. An easy way to implement this is by using the Sutherland line clipping algorithm in three dimensions. This algorithm is pretty widely used in polygon clipping in computer graphics applications.

The Sutherland clipping algorithm¹⁵ is widely used in computer graphics applications to clip polygons visible in a canvas. The idea is to have an efficient way to ignore the line segments that lie completely outside the canvas. If we extend the canvas to 3 dimensions, we get a box and instead of 2-D line segments we can deal with 3D line segments. Correlating line segments with the motion of molecules in one time step, and the 3D box with a box bounding our solid object, we can directly use the Sutherland algorithm for our problem.

Figure 5 illustrates the Sutherland algorithm in 2D. The 2D space is divided by the square (canvas) into 4 regions with respect to the canvas - LEFT, TOP, RIGHT and BOTTOM. Each point in the 2D space can now be located by using a binary representation in four bits one each for LEFT, TOP, RIGHT and BOTTOM. The appropriate bits are turned on depending on the location of the point (i.e. if the point is to the left and top of the square then the LEFT and TOP bits are 1 and the rest are 0). It is intuitive that if we get a nonzero value when we perform an “AND” operation on the binary representation of the end points of a line segment, the segment will never intersect the canvas.

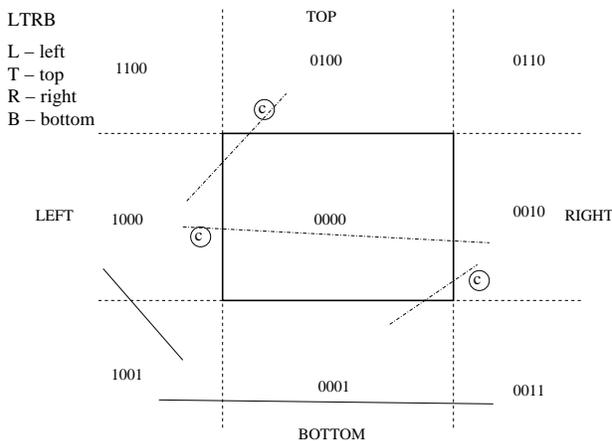


Fig. 5 Sutherland line clipping algorithm for a 2-D case. The lines marked with © should be clipped.

This concept is easily extended to 3 dimensions by adding two more bits - OUT and IN for the third dimension. Figure 6 illustrates the point. The “AND” operation is now performed over the 6-bit binary rep-

resentation but there is no increase in cost for this check. The cost increase is only in identifying the (binary) location of the particle in the 3D space.

O I L T R B

O – out
I – in
L – left
T – top
R – right
B – bottom

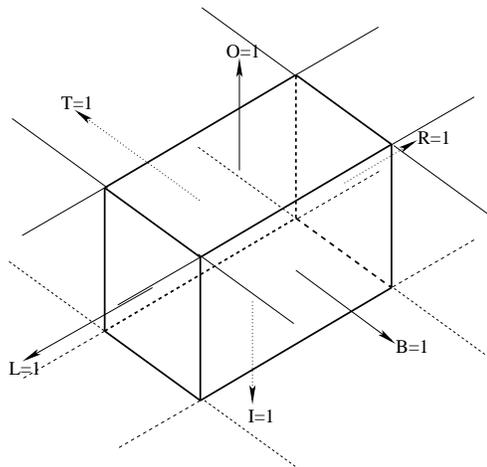


Fig. 6 Sutherland line clipping algorithm extended to 3 dimensions.

Once we are through the bounding box test and the line segment is found to intersect the bounding box, we do further tests to eliminate the molecules which may not intersect the solid body. If there is no collision (with the solid body), the molecule would go from position A to position B in the time interval Δt (refer Fig. 7). We use a parametric representation of the line segment AB.

$$\mathbf{X} = \mathbf{X}_A + \mathbf{V}t \quad (6)$$

where $\mathbf{V} = \mathbf{X}_B - \mathbf{X}_A$ is the fictitious velocity of the molecule such that the molecule will reach B from A in $t = 1$ unit. For finite AB, t can range only between 0 and 1.

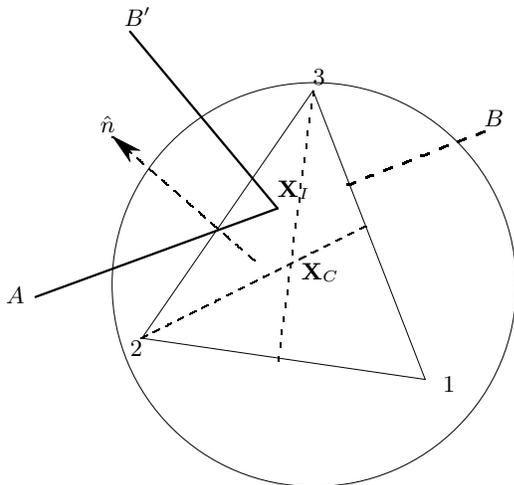


Fig. 7 A schematic of a molecule reflecting off a solid triangular surface.

The equation of an infinite plane formed by the three points \mathbf{X}_1 , \mathbf{X}_2 and \mathbf{X}_3 is

$$(\mathbf{X} - \mathbf{X}_1) \cdot \hat{\mathbf{n}} = 0 \quad (7)$$

where, \mathbf{X} is the vector defining the plane and $\hat{\mathbf{n}}$ is a unit normal to the plane calculated using Eq. 8.

$$\hat{\mathbf{n}} = \frac{(\mathbf{X}_1 - \mathbf{X}_2) \times (\mathbf{X}_3 - \mathbf{X}_2)}{|(\mathbf{X}_1 - \mathbf{X}_2) \times (\mathbf{X}_3 - \mathbf{X}_2)|} \quad (8)$$

Note that the normal to the plane need not be calculated at every time step. It may be calculated during the first iteration and stored for the rest of the simulation. The intersection point, \mathbf{X}_I of the segment with the infinite plane can be obtained by simultaneously solving Eqs. 9 and 10 for t^* and \mathbf{X}_I .

$$(\mathbf{X}_I - \mathbf{X}_1) \cdot \hat{\mathbf{n}} = 0 \quad (9)$$

$$\mathbf{X}_I = \mathbf{X}_A + \mathbf{V}t^* \quad (10)$$

The solution of the above equations is

$$t^* = \frac{(\mathbf{X}_A - \mathbf{X}_1) \cdot \hat{\mathbf{n}}}{\mathbf{V} \cdot \hat{\mathbf{n}}} \quad (11)$$

and \mathbf{X}_I may be obtained from Eq. 10 using t^* from Eq. 11. The finite segment, AB intersects the infinite plane only if $0 \leq t^* \leq 1$ The equality on either side meaning that \mathbf{X}_I is the same as \mathbf{X}_A or \mathbf{X}_B . If t^* does not satisfy the above condition, then we can say that the molecule will not hit the triangle. If, however, t^* satisfies the above condition then we need to find out if \mathbf{X}_I lies inside the triangle.

A quick elimination of a number of molecules *not* intersecting the triangle may be performed by checking if \mathbf{X}_I lies outside a circle enclosing the triangle. An obvious choice for this is the circumcircle, but we do not choose the circumcircle because it is computationally involving to obtain the center of a circumcircle in three dimensions. We choose a circle with a radius, $r = 2/3 \times$ the largest edge of the triangle, and its center at the centroid. This circle will completely enclose the triangle but will not exactly circumscribe it. The reasoning behind this is - the vertex of a triangle farthest from its centroid is at a distance of $2/3 \times$ the maximum of the three medians. Since the largest edge of a triangle is always greater than the largest median, the circle with radius r and center at the centroid of the triangle will completely enclose the triangle.

We calculate the centroid of the triangle and the radius of the enclosing circle described above during the first iteration. All \mathbf{X}_I s with $|\mathbf{X}_I - \mathbf{X}_C| > r$ will hit the infinite plane outside the circle and hence outside the triangle. Here, \mathbf{X}_C is the centroid of the triangle.

If \mathbf{X}_I lies inside the circle then there is no shortcut to figure out if it is inside the triangle or outside. Since the vertices of all the triangles are stored in a specific order (cyclic or anti-cyclic), the angles subtended by each edge (viz.. 1-2, 2-3 and 3-1) of the triangle at the point \mathbf{X}_I will all be either positive or negative if \mathbf{X}_I lies inside the triangle. If it lies outside the triangle then there will be a change of sign. Figure 8 illustrates the

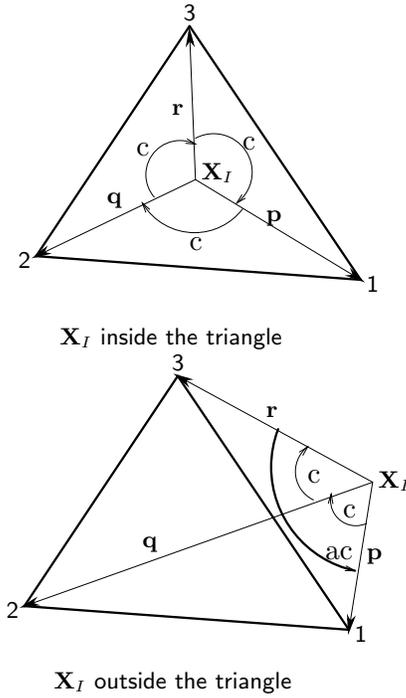


Fig. 8 An illustration showing that the angle subtended by all the edges have the same sign if \mathbf{X}_I is inside the triangle but they change sign if \mathbf{X}_I is outside. The angle subtended is positive if it is clockwise (represented by ‘c’), and negative if it is anti-clockwise (represented by ‘ac’).

above point. In fact, the sum of the subtended angles is equal to zero if \mathbf{X}_I is outside the triangle, but we don’t want to do an extra summation.

The change in the sign of the subtended angles can be perceived by looking at the directions of the cross products - $\mathbf{p} \times \mathbf{q}$, $\mathbf{q} \times \mathbf{r}$ and $\mathbf{r} \times \mathbf{p}$, where $\mathbf{p} = \mathbf{X}_1 - \mathbf{X}_I$, $\mathbf{q} = \mathbf{X}_2 - \mathbf{X}_I$ and $\mathbf{r} = \mathbf{X}_3 - \mathbf{X}_I$. If all of them point in one direction then \mathbf{X}_I lies inside the triangle otherwise it lies outside the triangle. If two vectors point in the same direction their dot product is positive; if they point in opposite directions then their dot product is negative. Note that $\mathbf{p} \times \mathbf{q}$, $\mathbf{q} \times \mathbf{r}$ and $\mathbf{r} \times \mathbf{p}$ are either parallel or anti-parallel depending on whether \mathbf{X}_I lies inside or outside the triangle since \mathbf{p} , \mathbf{q} and \mathbf{r} are coplanar. We check the dot products $(\mathbf{p} \times \mathbf{q}) \cdot (\mathbf{q} \times \mathbf{r})$ and $(\mathbf{q} \times \mathbf{r}) \cdot (\mathbf{r} \times \mathbf{p})$; if both are positive then we conclude that \mathbf{X}_I lies inside the triangle otherwise it lies outside the triangle.

If a molecule hits the solid surface then it is bounced off the surface specularly just like a light ray reflects off a mirror. This is fairly easy to implement if we write the velocity vector as a sum of the normal velocity (normal to the plane) and tangential velocity. The tangential direction is given by $\mathbf{V}_t = \mathbf{V} - \mathbf{V}_n$. For the reflected velocity, we just change the sign of the normal velocity while the tangential component remains unchanged. Hence, reflected velocity is $\mathbf{V}_t - \mathbf{V}_n$. The final position is calculated by moving the parti-

cle with the reflected velocity for the remaining time taking into consideration that there might be further collisions if the solid body is concave.

Model Test Cases

Two model shapes were tested for blast impact using our DSMC code. These are a box and an ‘I’ shaped beam (I-beam). The I-beam was chosen because it is a concave geometry which is challenging to model. A Cartesian grid is used to divide the rectangular domain. The particles are distributed randomly across the domain and their velocities are given by a Maxwellian distribution at the start of each run. All the particles inside the solid body (closed surface) are removed. This is done to avoid any information exchange into the solid body because of intermolecular collision. The solid and inflow boundary conditions are applied as explained in the previous sections. The following set of initial conditions (in SI units) were used for the cases presented here.

$$\rho_1 = 1.226 \quad ; \quad T_1 = 300 \quad ; \quad \rho_4 = 1.226 \quad ; \quad T_4 = 600$$

The shock wave generated has a Mach number of 1.18 and the temperature and pressure behind the wave are 351.9°K and 156561.9 N/m². The density behind the shock falls to 1.0246 kg/m³ for the chosen conditions. This is a relatively mild shock wave which is chosen purely for demonstration purpose.

The net force on the bodies is calculated by summing up the force due to each molecule which is obtained using first principles - the rate of change of the momentum of the particle. The loading on the box and the I-beam are shown in Figs. 9 and 10 respectively. A sharp jump in the force can be observed in the case of the box which corresponds to the shock wave hitting the front face. The points marked ‘1’ and ‘2’ in Fig. 10 correspond to the times when the shock wave hits the flanges and the main spar respectively of the I-beam. The pressure contour plots at these times are shown in Figs. 12 and 13.

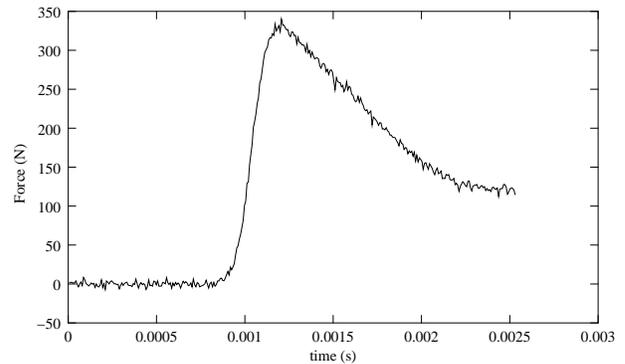


Fig. 9 Time history of the loading on the Box.

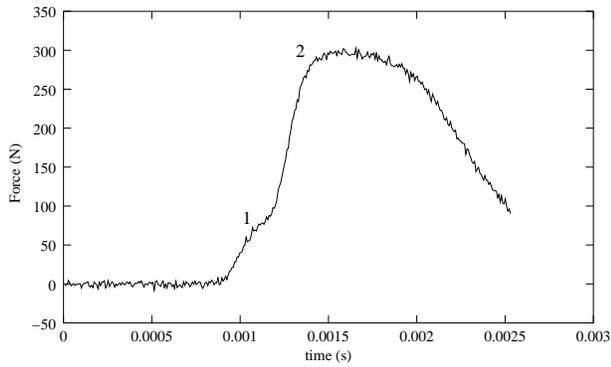


Fig. 10 Time history of the loading on the Ibeam.

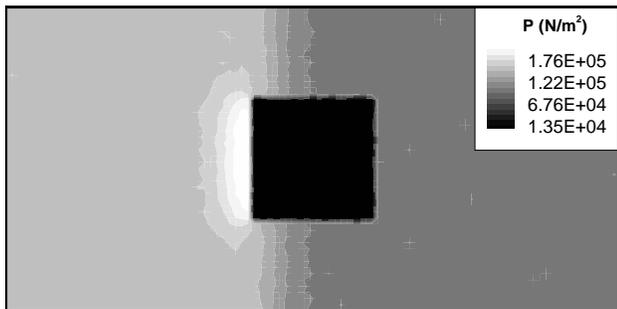


Fig. 11 Pressure plot when the shock wave hits the box.

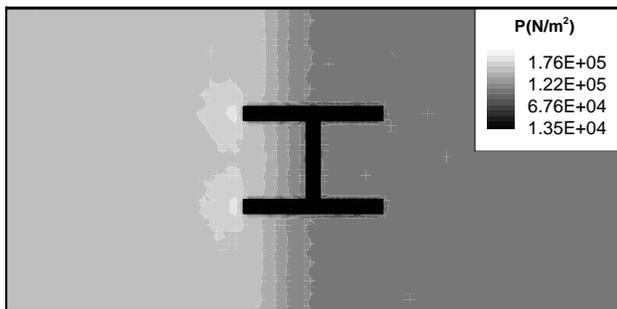


Fig. 12 Pressure plot when the shock wave hits the flanges of the I-beam; marked '1' in Fig. 10

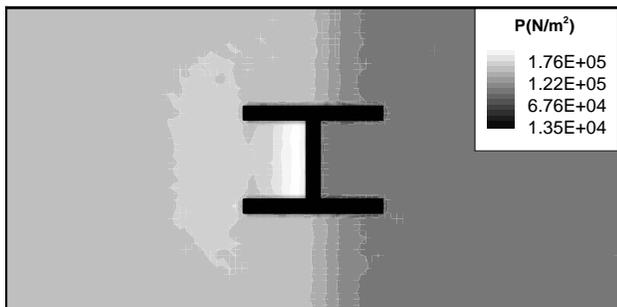


Fig. 13 Pressure plot when the shock wave hits the middle of the Ibeam; marked '2' in Fig. 10

Conclusions

The DSMC approach has been used for an uncoupled analysis of the blast-impact problem. DSMC is applied in an efficient way in the continuum regime by limiting the number of collisions per time step. Our DSMC code is validated against the shock-tube problem. A new inflow boundary condition is developed which can simulate the shock wave without the corresponding expansion wave. This reduces the size of the computational domain and hence the cost associated with it. The solid boundary condition is implemented on a molecular basis in a very generic fashion. This boundary condition can easily and efficiently handle arbitrary-shaped geometries. The load definitions on two model geometric shapes are presented for a chosen set of conditions. DSMC has been shown to be capable of simulating the load-definition problem. We intend to do the coupled analysis in future where the solid body deformation shall be modeled by a finite element analysis.

Acknowledgement

The authors would like to acknowledge the Protective Technology Center¹⁶ at the Pennsylvania State university for sponsoring this research.

References

- ¹J. K. Wright, *Shock Tubes*, John Wiley & Sons INC, 1961.
- ²W. Bleakney, D. R. White, and W. C. Griffith, "Measurements of Diffraction of Shock Waves and Resulting Loading on Structures," *Journal of Applied Mechanics*, Vol. 17, Dec. 1950, pp. 439–445.
- ³L. N. Long, M. Kamon, T. S. Chyczewski, and J. Myczkowski, "A Deterministic Parallel Algorithm to Solve a Model Boltzmann Equation (BGK)," *Computing Systems in Engineering*, Vol. 3, No. 1-4, 1992, pp. 337–345.
- ⁴Lord Kelvin, "Nineteenth Century Clouds over the Dynamical Theory of Heat and Light," *Philosophical Magazine*, Vol. 2, No. 6, 1901, pp. 1–40.
- ⁵G. A. Bird, "Direct Simulation of the Boltzmann Equation," *Physics of Fluids*, Vol. 13, Nov. 1970.
- ⁶G. A. Bird, *Molecular Gas Dynamics*, Clarendon Press, Oxford, 1976.
- ⁷G. A. Bird, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Oxford Science Publications, 2nd ed., 1994.
- ⁸L. N. Long, "Navier Stokes and Monte Carlo Results for Hypersonic Flows," *AIAA Journal*, Vol. 29, No. 2, Feb. 1991.
- ⁹L. N. Long and J. B. Anderson, "The Simulation of Detonations using a Monte Carlo Method," *RGD Conference, Sydney Australia*, 2000.
- ¹⁰J. B. Anderson and L. N. Long, "Direct Simulation of Pathological Detonations," *RGD Conference Whistler, British Columbia, Canada*, 2002.
- ¹¹P. J. Morris, S. B. Esfahaani, and Y. P. Liew, "Simulation of Blast Loads on Arbitrary Geometry Structures," *The 17th International Symposium on Military Aspects of Blast and Shock*, Jun. 2002.
- ¹²L. N. Long, "A Nonconservative Nonlinear Flowfield Splitting Method for 3-D Unsteady Fluid Dynamics," *AIAA Aerocoustics Conference*, Jun. 2000.
- ¹³C. L. Merkle, H. William Behrens, and Robert D. Hughes, "Application of the Monte-Carlo Simulation Procedure in the Near Continuum Regime," *Presented at the Twelfth Symposium on Rarefied Gas Dynamics*, Jul. 1980.

¹⁴J. D. Anderson Jr., *Modern Compressible Flow with Historical Perspective*, McGraw Hill Professional Publishing, 2nd ed., Oct. 1989.

¹⁵I. E. Sutherland, R. F. Sproull, and R. Shumacker, "A Characterization of Ten Hidden Surface Algorithms," *ACM Computing Surveys*, Vol. 6, No. 1, 1974, pp. 1-55.

¹⁶<http://www.ptc.psu.edu/>.