The DS2V/3V Program Suite for DSMC Calculations

G. A. Bird

Articles you may be interested in

Compressible inviscid instability of rapidly expanding spherical material interfaces
Phys. Fluids 24, 034106 (2012); 10.1063/1.3689183

Simulations of Richtmyer–Meshkov instabilities in planar shock-tube experiments
Phys. Fluids 23, 034106 (2011); 10.1063/1.3555635

Comparison between Navier-Stokes and DSMC Calculations for Low Reynolds Number Slip Flow Past a Confined Microsphere
AIP Conf. Proc. 762, 701 (2005); 10.1063/1.1941617

Assessment of thermodynamic models for dense gas dynamics
Phys. Fluids 16, 3878 (2004); 10.1063/1.1786791

Pressure boundary treatment in internal gas flows at subsonic speed using the DSMC method
AIP Conf. Proc. 585, 408 (2001); 10.1063/1.1407590

The DS2V/3V Program Suite for DSMC Calculations

G.A. Bird

GAB Consulting Pty Ltd
144/110 Sussex Street, Sydney NSW 2000, Australia

Abstract. There is a need for general purpose DSMC codes that can be readily applied by non-specialist users to a wide variety of practical problems. The specifications that should be met by such programs are outlined and the currently available programs are noted. The DS2V program for two-dimensional and axially-symmetric flows and the DS3V program for three-dimensional flows are described in some detail. The "V" in the program names is to indicate the interactive visual characteristic of the programs. Both programs run in a time-accurate mode and the flow sampling may be of the unsteady flow or, if a steady flow is established at sufficiently large times, of the steady flow. The gas model includes internal degrees of freedom, gas phase chemical reactions, and surface reactions. The initial state may include flow discontinuities that permit the study of shock tube flows and free shear flows. Solid surfaces may move in their own plane, a typical application being a rotating body in an axially-symmetric flow. Alternatively, a surface may move in a normal direction to generate a moving shock wave for diffraction studies. Flow boundaries may be arbitrary combinations of solid surfaces and specified flows. Other options include periodic boundaries and constant pressure boundaries that have been developed especially for MEMS applications.

INTRODUCTION

There is a growing requirement for the DSMC method to be applied to practical problems by users who are familiar with the application, but who have little interest in or knowledge of rarefied gas dynamics. It is no more reasonable to expect such a user to write his own DSMC program than it is to expect an author to write his own word processing program! An increasingly important issue with regard to the DSMC method is therefore the degree of difficulty that is experienced by non-specialist users in obtaining programs that provide reliable results for their problems of interest. From the point of view of the potential user, the immediate concerns relate to the cost and availability of programs. Should these be satisfactorily resolved, the ease of use of the programs and the extent to which they provide information on, or even attempt to enforce, the validity of the solutions are matters of critical importance. The degree of approximation of the underlying mathematical and physical models is also of concern, as is the adequacy of the available database of physical properties.

If the problem of interest has a very simple geometry, useful results can sometimes be obtained by the demonstration programs that were included as FORTRAN source code on a floppy disk that was included in the first printing of the 1994 monograph by the author [1]. These programs can now be freely downloaded from gab.com.au, as can the executable of the earlier general DSMC program DS2G. The DS2G executable is for Windows PCs and the data input menus are typical of those available in the "DOS era". The DS2G program employs "ray tracing" logic for the movement of the simulated molecules and is therefore slower than the more modern programs that generally employ some sort of "rectangular reference" scheme. It also has a fixed time step rather than the variable time steps that are employed in more modern programs. However, it incorporates the "transient sub-cells" that have proved to be invaluable in the DS2V/3V programs that are discussed in the later sections of this paper.

SMILE [2] is a modern general DSMC program that has a number of users and a history of well validated applications. It is a commercial program, but there is little information on its availability and cost. The DAC program [3] was developed at the Johnson Space Center and is evidently available for restricted distribution. The acronyms SUPREM and HAWK also appear when a search is made for general DSMC programs.

SPECIFICATIONS FOR THE IDEAL GENERAL DSMC PROGRAM

It is suggested that the capabilities of the existing and any new general DSMC programs should be evaluated against the following specification of the "ideal program"

Capability

The range of application of a program in terms of the lower Knudsen number limit will always be restricted by the performance of the computers that are available at any particular time and place. Advances are likely to increasingly depend on parallel computation rather than on increases in CPU speed and the program should be capable of parallel operation. The program should be capable of application to any problem that falls within the capabilities of the contemporary computers. These problems will involve internal or external flows in one, two or three-dimensions and will be either large-time steady flows, or time-accurate unsteady flows. The program should incorporate a range of realistic models for intermolecular collisions, internal degrees of freedom, gas phase chemical reactions, and gas-surface interaction and reactions. It should be possible to simulate the effects of a gravitational fields and similar body forces. The initial conditions should allow the specification of flow discontinuities and the boundaries should allow for periodic flows, constant pressure boundaries, and moving surfaces.

Validity

The programs should reproduce the solutions for all simple test cases that are amenable to exact analysis. The results from the program should be in agreement with any valid experimental measurements.

Apart from any influence exerted by the magnitude of the computer resources that are devoted to the problem, the degree of agreement with the test cases should not be affected by the settings of any computational variables that are under the control of the user. Benchmarking must be objective rather than subjective. Not only should the program automatically set the computational parameters to near optimal values, it should be able to internally check the quality of the computation. The probable error in the results should be reported to the user and, should the quality of the calculation be outside specified limits, the program should automatically abort the calculation.

Availability

The program should be readily available and, for use on a Windows PC, the cost should be within the established price range of specialist application programs. The program should be capable of being ported to other computers and operating systems.

Usability

The program should have a modern interactive graphical user interface. The data input should be through a fully specified file that can be generated by interactive menus within the program. There should be a user manual with representative demonstration cases. There should also be a help system within the menu structure that provides the necessary information on the data items. The computational details or the log of the calculation, the flow contours, and the distribution of the surface properties should be displayed continuously as the calculation proceeds. Calculations should be interactive in that the user has control over the properties that are being displayed and can continuously interrogate the values of all properties at any point within the flow.

The user should be able to obtain useful results for his problem on the first day that he acquires the programs. Most importantly, apart from the error due to the improper positioning of the flow boundaries, the user should not be able to obtain a bad result.

DS2V/3V GEOMETRY MODEL

The then existing grid systems or geometry models for two-dimensional flows were discussed in Reference [1]. With regard to three-dimensional flows, it was stated that the "optimal grid scheme for engineering studies of complex three-dimensional flows of rarefied gases has yet to be determined". Version 1 of the DS2V program employed a combination of rectangular cells in the outer region and body fitted cells near to surfaces. This model would have been extremely difficult to scale to three dimensions and it was replaced in version 2 by the model shown in Fig. 1.

A "bounding rectangle" is divided into rectangular "divisions" and the divisions are, in turn, divided into rectangular "elements". The number of divisions affects the efficiency rather than the quality of a simulation, but should approximate the number of partitions into which a small disturbance flow would have to be divided in order to provide a reasonable visual resolution of the flowfield. The number of elements per division is such that the number of elements within the flowfield approximates the number of simulated molecules.
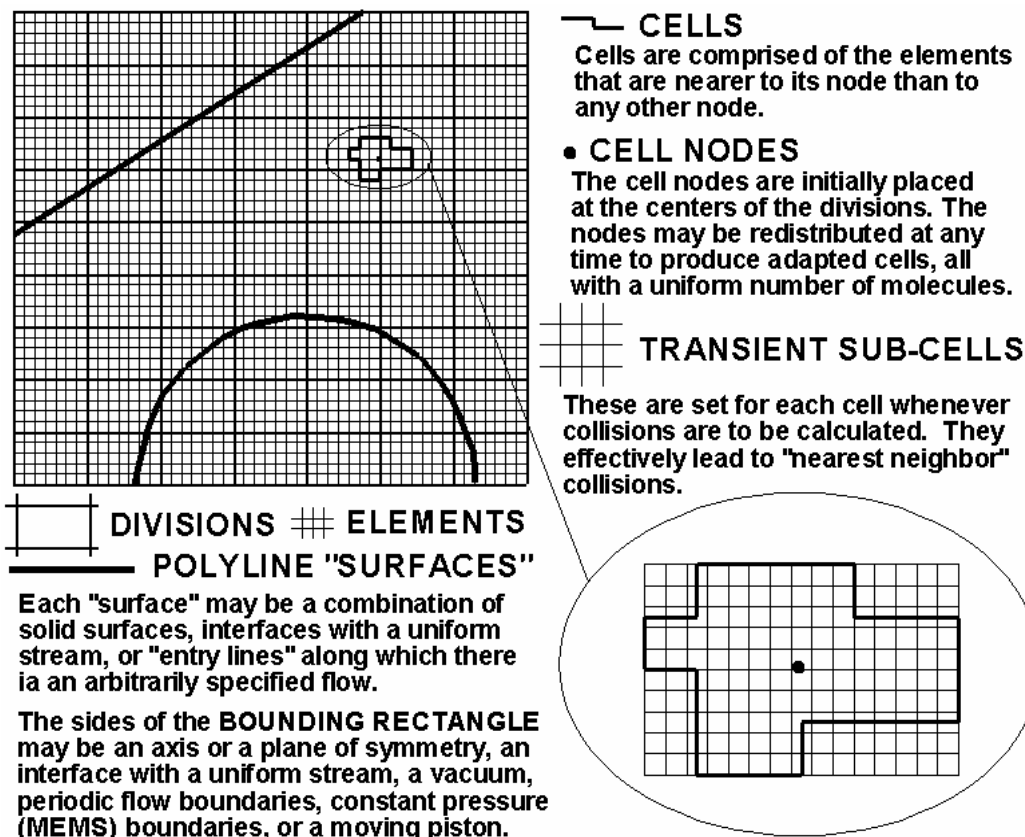

FIGURE 1. The geometry model for version 2 of the DS2V program.

Any number of "unconnected surfaces" may be defined and these are either closed surfaces within the flow or open surfaces that must start and terminate on the bounding rectangle of the flowfield. There is a restriction on the geometry in that the thickness of a closed surface must be sufficiently large that a single element cannot have isolated regions within the flow on both sides of the surface. A surface may be defined by a polyline, in which case each element is a "sampling interval". Alternatively, straight line segments may be defined by their endpoints and circular segments by endpoints together with the location of the center. The number of sampling intervals along each segment is then specified separately. The resolution of the surface property output is set by the size of the intervals. The "type" of the surfaces is defined for arbitrarily sized "groups" of sampling intervals. A surface interval group may be a solid surface, a specified inflow, or a boundary with one of the two streams that may be specified. Stream surfaces may cut off regions of uniform flow in the corners of the rectangular flowfield.

The divisions that contain surface segments are defined as "level 0" divisions and the surface segments are indexed to these divisions. A molecule must be tested for collisions with a surface segment if any part of its trajectory falls within a level 0 division. The divisions that are within the flow and have a face in common with a level 0 division are called "level 1" divisions, and so on. Divisions that are entirely outside the computational flowfield are "level -1" divisions. The distance that a molecule can move in a single time step is restricted to the extent that it cannot pass through more than three division levels. This means that the possibility of an interaction with a surface need not be considered within a time step if the molecule is initially in a division at level 3 or above. The computational speed of the "move" routines in the DS2V program is many times faster than the corresponding routines in the earlier G2 and DS2G programs.

543

Having determined the end-point of a trajectory segment, it is necessary to determine the flow "cell" in which it lies. The code number of the new element is readily determined and the only item that is stored for every element is the code number of the cell in which it lies. An element is in a given cell when it is nearer to the "node" of that cell than to any other node. Initially, the cell nodes are set at the centers of the divisions that are entirely within the flow and also in divisions that are largely within the flow. Whenever a calculation is restarted after having been stopped, there is an option to adapt the cells. The desired number of molecules per cell is then specified, the total number of nodes is set to the number of simulated molecules divided by the desired number, and the cell nodes are then distributed through the flow such that the probability of a given node location is proportional to the average number density in the division at that location. When assembling the nearest elements in order to make up a cell, it is of course essential to check whether there are any surfaces between the cell node and the element under consideration.

Traditionally, the cells are employed for both the sampling of the flow properties and for the determination of the number of trial collision pairs over the local time step within the collision routine. However, the best results are obtained if the number of molecules per cell is of the order of five to ten and, because a typical computation now contains at least a million simulated molecules, the number of sampling cells has become excessive. Not only are the output files extremely large, but the cell related storage within the program can be larger than the storage associated with the molecules. The solution is to employ separate "collision cells" and "sampling cells" and this is being implemented in version 2.2 of DS2V and version 1.3 of DS3V. The collision cells will be adapted to a fixed number of eight molecules per cell. The only items that need be stored for each collision cell are the number of molecules within the cell, the maximum value of the product of collision cross-section and relative velocity, the cell volume, and the cell time. The traditional expression for the number of trial collision pairs has also contained the average number of molecules in the cell $<N>$, but it can be shown that the mean value of $N(N\text{-}1)$ is identical to the mean value of $N<N>$. It was the requirement to have the average number of molecules in the cell that originally caused the collisions to be associated with the sampling cells. The two sets of cells will be made independent of one another and will be set by separate passes through the existing cell adaption routines. This means that two sets of cell numbers will be stored for each element.

The quality of a computation at the location of a flowfield cell is set primarily by the ratio of the mean separation of the collision pairs in a cell to the mean free path of the molecules within that cell, or "mcs/mfp" ratio. Therefore, having chosen the first molecule of a potential collision pair at random from those in the (collision) cell, it is not sufficient to choose the second molecule from an arbitrary location within the cell. Instead, the nearest molecule must be chosen and "transient sub-cells" have been employed to achieve this. A rectangular grid is placed over the cell and the molecules are indexed to the grid. The program considers the cell volume relative to the volume of the enveloping rectangular grid and the number of molecules in the cell. The grid size is chosen such that there is approximately one molecule per grid element. If there is more than one molecule in the sub-cell, the second molecule is chosen from these. Otherwise, the "layers" of sub-cells surrounding this sub-cell are progressively tested until a molecule is found and, if there are no molecules in the first six layers, the molecule is chosen from anywhere in the cell. Each layer is tested through a linear look-up table with a random entry point.

The importance of the mcs/mfp ratio cannot be over-estimated because it enables the calculation to be "self validating". With all effects of cell size having been eliminated and with the program automatically setting a sufficiently small time step, it provides a single parameter measure of the quality of the computation. This means that the percentage error in the result can be reliably predicted and the program can be set to abort the calculation if the number of simulated molecules is insufficient to provide the required accuracy.

The DAC program [3] employs "virtual sub-cells" as an alternative to transient sub-cells. The distance from the first molecule of all the other molecules in the cell is determined and the nearest is selected. This is a simpler procedure, but the computation time is proportional to the square of the number of molecules in the cell. Virtual sub-cells have now been implemented as an option in the DS2V/3V programs and it has been found that they are faster and more effective than transient sub-cells when the number of molecules in the cell is less than about 35. The computational time penalty associated with transient sub-cells in a typical application was found to be about 12% and was near uniform over the normal range of molecule numbers per cell. That associated with virtual sub-cells was about 5% for 8 molecules per cell, 10% for 32 molecules per cell and almost 100% for 256 molecules per cell. In addition, virtual sub-cells produced an additional reduction in the mean spacing between colliding molecules of 17%. The programs now automatically select virtual sub-cells for less than forty molecules per cell and transient sub-cells when the number of molecules per cell exceeds this number. This means that transient sub-cells will rarely be activated when the separate sampling and collision cells have been implemented and the target number of molecules per cell in an adaption is automatically set to eight.

A problem that arises with probabilistic nearest-neighbor collisions is that there is a high probability that a collision of a particular molecule will be with the same collision partner that was involved in its immediately preceding collision. This is physically impossible and would not occur in a deterministic simulation. The solution is to store for each molecule the code number of the molecule with which it last collided and to forbid successive collisions with that molecule. The programs have the capability to store the preceding six collision partners, but it was found that there was no need to go further than the immediately preceding collision. Note that it is not sufficient to forbid successive collisions only within a time step because this leads to deteriorating accuracy as the time step is made very small.

The extension of the two-dimensional geometry model that is shown in Fig. 1 to three-dimensional flows is straightforward. The rectangular outer boundaries, divisions, and elements simply become rectangular parallelepipeds. The most significant change is that the polyline elements or line segments that define the surfaces in two dimensions must be replaced by triangular elements. This means that it is no longer possible for the user to define all the data for a problem through a single set of interactive menus. A separate surface definition data file had to be introduced and Version 1 of the DS3V program requires the user to either write a program for the surface triangulation or to obtain a triangulation from an expensive grid generation program. The size of the triangles determines the resolution of the surface quantities in the output display and files and it is essential for the user to have control over the triangle size. The Rhinoceros program [] is a relatively inexpensive three-dimensional modeling program that provides the necessary functionality and can output the surface definition as a list of "raw" triangle coordinates. Version 2 of DS3V will convert a raw triangle data file to the format required by the program and will then plot the triangles in order to permit the user to define the inside and outside of the shapes. It will also provide menus that allow the user to specify whether groups of triangles are solid surfaces or flow entry surfaces and to interactively specify the properties of the triangles in the groups.

DS2V/3V DATA SPECIFICATION AND PHYSICAL MODELS

The user must first set the size of the bounding rectangle or rectangular parallelepiped of the region to be simulated. The fixed geometry may appear to be a disadvantage but, as noted earlier, bounding surfaces (that include stream interfaces) may be added to achieve almost any desired flowfield shape. A poor result may be obtained if the user makes a bad choice of boundaries, but this should be the only opportunity for a user to make a poor calculation.

The size of the division and element grids is the next data item in the current versions of the programs and this precedes the specification of the molecular properties and the reference state of the gas. This order will be reversed in the next versions and the user will be asked to instead nominate the number of Megabytes that will be available for the running of the program. There will then be sufficient information for the program to make an intelligent automatic choice of division and element size.

The programs employ the variable hard sphere model for intermolecular collisions. The VSS model is to be added so that simulations of gas mixtures may be made with the correct Schmidt number. This model was implemented in the later versions of the older DS2G program and its current absence from the new codes is because the DS2V program was originally developed from a simplified DSWT "wind tunnel" code. The classical Larsen-Borgnakke model is employed for rotational degrees of freedom, while the quantum Larsen-Borgnakke model is employed for the vibrational modes. This means that the discrete vibrational levels are stored for each molecule rather than just a continuously distributed vibrational energy. There is a case for the quantum model to be extended to rotation, especially if bound-bound thermal radiation was to be added to the program capabilities. Chemical reactions are currently modeled through the conversion of the continuum rate constants to reactive cross-sections through a quasi-equilibrium theory. It is planned to optionally model dissociation reactions as vibrational excitation to the dissociation level and this ability to directly model physically realistic "event-driven" processes is a major advantage of the physical simulation approach. There will also be provision for the direct input of reactive cross-sections, or steric factors, for all reactions. Gas-surface interactions are currently modeled through diffuse reflection with the option of a specified fraction of specular reflection. The next versions will include the more realistic CLL model as an additional option. This model is also standard in the DS2G code, but has been little used because almost no information is available on the accommodation coefficients that should be employed for space-related applications. Chemical reactions that are a consequence of the gas-surface interaction have already been implemented and there are options for outgassing and adsorbing surfaces.

The description of the geometry model in the preceding section has already introduced some of the capabilities of the codes. Other capabilities are implicit in the range of options within the menus for the specification of solid and inflow surfaces. As an alternative to the specification of the temperature distribution over the surface, it may be specified as an adiabatic (zero heat transfer) surface. The output then reports the temperature distribution that corresponds to the adiabatic condition. The surfaces may move in their own plane and two components of velocity may be specified. In the case of two-dimensional plane flows, one component is in the plane of the flow and, for a cylinder with axis normal to the plane, this can be used to apply a spin to the cylinder. On the other hand, a velocity normal to the plane can be used to simulate a swept wing of infinite span. The only in-plane velocity that is physically realistic in axially-symmetric flows is the normal component and this is specified as an angular velocity for bodies that spin about the axis.

The options for the specification of bounding planes and for an internal separation boundary also influence the range of problems that may be simulated. These include periodic boundaries for infinite or cyclical flows and constant pressure boundaries that allow a mass flow through an internal flow to be established by the pressure difference without any velocities having to be specified at the boundaries. One boundary may act as a piston and move with a uniform speed into the flow and generate a shock wave. Waves can also be generated at the initial separation plane if it separates gases at different pressures. Alternatively, the plane may separate gases at the same pressure, but with a velocity or temperature discontinuity.

The user must specify whether the output sampling is to be for a continuing unsteady flow or for a flow that reaches a steady state a sufficiently large time. There is also a requirement in the current versions for the user to set a reference time that influences the intervals for flow sampling and output. This item has caused difficulties for some users and the next versions will automatically set recommended values, but will allow adjustments by experienced users.

FUTURE DEVELOPMENTS AND CONCLUDING REMARKS

The current versions of the DS2V/3V codes come much closer than any of the earlier programs by the author to meeting the above specifications for the ideal general DSMC program. Current efforts are directed towards bringing the next full versions to near complete conformance with the specifications. While these specifications refer to "a program", there are currently separate programs for two and three dimensional flows. The DS2V program can also deal with one-dimensional flows through the setting of a single division in one direction with periodic boundaries on the longer sides, but it would be more efficient to not move the molecules at all in the redundant dimension. It is planned to amalgamate DS2V and DS3V into a single "Visual DSMC" code and to include formal one-dimensional and zero-dimensional (homogeneous gas) options. The resulting code would be about sixty thousand lines in length and this would grow towards one hundred thousand as additional capabilities were added to further increase the range and variety of applications.

The large samples and consequent low scatter that is characteristic of current results permit reliable automatic testing to distinguish between regions of viscous and inviscid flow. The collision rate in inviscid or isentropic regions can then be reduced to the level required to just maintain an equilibrium distribution function. The time step can be increased in these regions in order to also reduce the computation time for molecule moves. This will permit two-dimensional calculations to be extended further into the continuum regime.

The shortcomings of the current programs that are difficult to remedy are those that relate to issues of portability between computer platforms and perhaps to the adequacy of the documentation and support. These largely reflect the restrictions that are inherent in large code development by a single person.

ACKNOWLEDGEMENT

REFERENCES

1. Bird, G.A., "Molecular Gas Dynamics and the Direct Simulation of Gas Flows," Oxford University Press, Oxford (1994).
2. Ivanov, M.S., Markelov, G.N., and Gimelshein, S.F., "Statistical Simulation of Reactive Rarefied Flows: Numerical Approach and Applications," AIAA Paper 98-2669, June 1998.
3. LeBeau, G.J., "A User Guide for the DSMC Analysis Code (DAC) Software for Simulating Rarefied Gas Dynamic Environments," Revision DAC97-G, Jan. 2002.