THE USE OF COMPUTER GRAPHICS IN THE NUMERICAL SIMULATION OF DETONICS AND PENETRATION MECHANICS

I G Cullis
P A Fozard
Royal Armament Research and Development Establishment
Fort Halstead
Sevenoaks
UK

Summary: This paper describes the development of various computer graphics software packages used at RARDE to interpret the results from numerical simulation of problems in detonics and penetration mechanics. The development and use of animation techniques, in particular, is shown to be a powerful tool in understanding the time dependant physical processes associated with a problem as well as identifying the origin and nature of numerical instabilities within a calculation. Finally, recent work to develop graphics code that visualises free bodies in two and three dimensions from an eulerian database is described.

1. INTRODUCTION

History is punctuated by scientific discovery having a profound effect on human civilisation. Nowhere has this been more dramatic than in the art of warfare, where the introduction of the longbow, gunpowder, and artillery to name but a few changed the course of history. Exploitation of such scientific discoveries, leading to rapid progress in a given technology, of course relies upon a sound understanding of the physical principles on which the new technology or model is based. This applies equally well to the development of adequate defence or forecasting the weather.

As our understanding of the physical world improves our past simple linear physical models are no longer sufficient to interpret our experiments. To make matters worse, our intuitive feel for the physical world fails with the introduction of non linear physics into our models. In this situation we are forced to turn to theoretical techniques. Current research in conventional defence is one area already at this watershed, since the study of detonics and penetration mechanics are dominated by non linear physical processes. The interaction of an explosive with a metal for example is characterised by transient states of extreme density and energy associated with shock waves.

The data needed today to improve our understanding of target defeat mechanisms is, therefore, difficult if not impossible to obtain experimentally. As a result many experiments limit the data acquisition to meet specific objectives. This limited scope adds little to our basic knowledge of the phenomena involved. Often, of course, the sheer scale of the experiment - eg explosive blast wave measurements-precludes extensive instrumentation. Within the warheads division at RARDE, therefore, we have been forced to turn to theoretical techniques to meet our research objectives in hypervelocity penetration, warheads and detonics. The main theoretical tools used in these studies are eulerian and lagrangian hydrocodes, that solve the equations of continuum mechanics. The immense amounts of data contained in a single hydrocode calculation cannot be assimilated quickly and accurately without the use of relevant graphics techniques. This paper describes the development at RARDE of the various graphics capabilities needed to interpret our theoretical studies.

2. NUMERICAL SIMULATIONS

2.1 Physics Requirements

The interaction of a material with an explosive can involve the whole spectrum of material response from hydrodynamic to elastic, over a wide temperature range. The physics of the interaction is dominated by the non linear physics of shock waves and is highly transient in nature (typical time period 100 μseconds). Under these conditions a material experiences severe and rapid distortions that generate significant flow fields. Pressures in excess of 400kbar, with associated compressions of 10-15%, and energies equivalent to 2000k, are generated in a metal in contact with a detonating explosive. The resultant material flow can be characterised by particle velocities of between 2-10km/s (mm/μsec).

It is interesting to observe that in the world of ballistics and detonics a "low" pressure is 1000bar, and a "low" velocity is 500m/s (10 times the recent hurricane windspeed). Attempting, therefore, to solve the continuum equations across this complete range of physical phenomena is a daunting task. Fortunately, the transient nature of the physics we are interested in allows us to assume the flow field to be isentropic and inviscid an approximation that greatly simplifies the nature of the problem.

2.2 Hydrocode Techniques

Over the past fifteen years two numerical approaches to the solution of these impulsive loading problems have been progressively developed. The power of a hydrocode lies in its ability to treat complex geometries and loading states in two and three dimensional situations without the need of various simplifications. In addition by being able to define uniform materials with precise properties, hydrocodes provide a powerful tool for parametric studies to compare theory with experiment and derive simple but rigorous physical models for predicting warhead performance.

The majority of hydrocodes use a lagrange formulation, ie the numerical mesh is embedded in the material and distorts as the material deforms. For simulations involving significant material flow the eulerian approach is much more successful. In an eulerian hydrocode the mesh remains fixed and material moves through it. Unlike lagrange codes, therefore, an eulerian computational cell can contain more than one material, each in a different thermodynamic state.

Both of these numerical techniques have been successfully used in many areas of ballistics and detonics research and development (for examples see Anderson 86, Cullis 86, Zukas 82.

For the majority of problems of interest to the Warheads Division at RARDE rapid material flow is the dominant feature. Most of our hydrocode research and development has, therefore, centred on eulerian techniques. The work horse code within the division is the USAF code Hull, acquired in 1978. During the past nine years we have extensively developed the code, particularly in respect of the differencing scheme and advanced colour graphics techniques. This work has gone hand in hand with developments by other users within the Hull group, mainly in the USA. The code has thus been exercised over an increasing range of problems, from nuclear blast simulation through to impact calculations sensitive to material strength. The resulting piece of software whilst not numerically perfect is nevertheless stable and can adequately treat most physical situations encountered in elasti-plastic-hydrodynamic flow phenomena associated with impulsive loading.

2.3 Algorithm Requirement

In most eulerian codes the equations of motion are solved, in conjuction with an equation of state, in two stages, a lagrange first step followed by a remapping, or advection step, which restores the distorted mesh back to its original configuration.

This advection step is by far the most important feature of an eulerian code, since the methodology employed must be capable of accurately defining and tracking material boundaries as they move through the mesh.

In detonics calculations this requirement is much more rigorous since, it must not only define metal/air interfaces accurately but must also be able to define and propagate discontinuities in density and pressure within a single material due to the formation of shocks and rarefactions. The algorithm must also resolve and propagate the complex elasto-plastic wave structures in those problems where strength effects are important. It must achieve this without introducing significant numerical dispersion and dissipation. Equally of course the main solution step must not introduce non-physical effects often associated with shock waves.

The resultant algorithm, therefore, represents a careful and pragmatic balance between solving the problem exactly and minimising computer time.

The obvious requirements of graphics techniques, in our research environment, therefore must accurately resolve the important physical features in a calculation; identify possible numerical effects, including instabilities and errors within the mesh, and achieve this with minimal computational effort.

2.4 Data Generation

Before discussing the graphics techniques employed in our calculations it is useful to outline the amounts of data that have to be assimilated in our calculations. The multi material nature of the eulerian cell implies that the computational cell has to carry information relating not only to global cell parameters but also species partition of mass, volume energy, stress and strain and any other material histories.

For a relatively simple penetration calculation using three materials the number of variable per cell (NH) is 17 in a 2D axisymmetric calculation. For a warhead simulation, with five materials NH increases to 25. The total number of variables of course depends on the number of cells in the mesh. Our simple penetration calculation may typically have a mesh of 100 x 250 cells. A single snapshot of the solution therefore generates 425kw of data. In three dimensions the extra dimension rapidly increases this to over 12MW (see Table 1).

	2D		3D	
Cell Size	0.5mm	1mm	1mm	
No X cells	80	100	100	
No Y cells	165	250	34	
No Z cells			140	
No materials	5	3	3	5
NH	23	17	20	26
Dataset Size (KWords)	575	425	9,520	12,376

TABLE 1: Hull Hydrocode: Data Generation

The tremendous size of these datasets, particularly in three dimensional calculations, implies significant amounts of I/O will be needed during the calculation and in post processing the results. The data storage requirement is also non trivial! It is quite clear, therefore, that to achieve any significant calculation throughput a supercomputer such as a CRAY is essential, since hydrocodes place severe demands on all aspects of a machines performance.

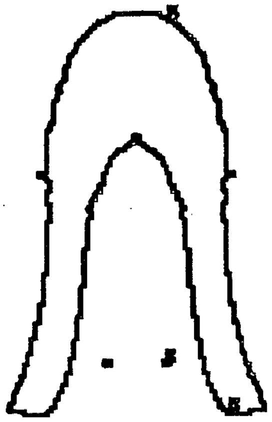
3. GRAPHICS TECHNIQUES

Wide ranging graphics capabilities are a pre-requisite for effectively processing and assimilating large volumes of data, whatever its source. Nevertheless the rapid assimilation of perhaps 70MWords of hydrocode output from a single 3D calculation is a complex problem. The large amounts of data generated also quickly fill up available disc space. The user is thus limited to the number of calculations that can be performed, as well as the number of dumps or snapshots in a calculation.

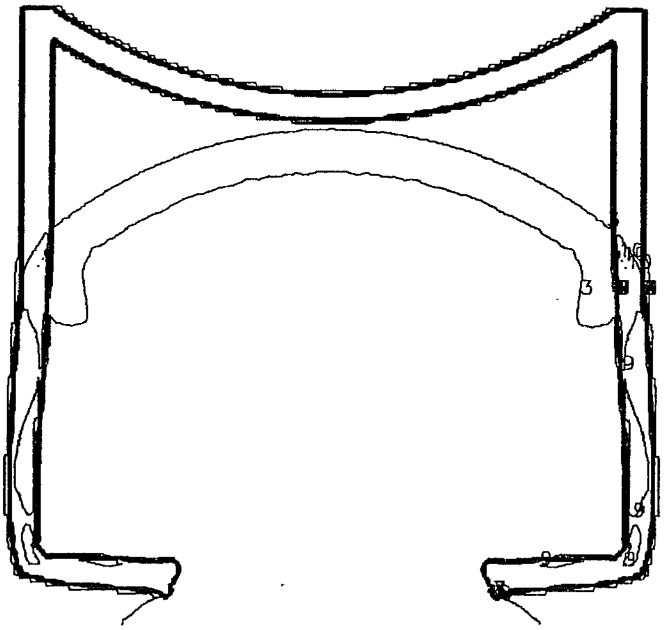
Standard contour plotting techniques can supply a useful quick "first look" at a calculation. We use these plots, in conjunction with a printed output file, to supply data in the form of material interfaces at various problem times. As an example, the explosive deformation of a curved iron dish to form a projectile travelling at 2km/s is shown in figure 1 at 7 and 110 μseconds after detonation. The early time plot shows case and liner interfaces together with leading edge of the detonation front. Details of the various shock waves in the problem are not resolved because of the number and range of the specified contours. This approach has three advantages. Firstly it identifies the time range over which a given physical process is important. Secondly it identifies parameter ranges for more detailed plotting. Finally non regular interfaces can often indicate numerical problems within the calculation that may need addressing before detailed graphical analyses can take place.

Very often, however, the ranges of parameters in different materials conflict, for example the pressure distributions in a metal and a gas. Whilst overlaying multiple plots with many contours, one for each material can sometimes produce passable results, the interpretation of details and contour identification can become almost impossible. This is particularly true when modelling the complex stress wave distributions set up in a metal block by a detonation wave. Figure 2 aptly illustrates this for aluminium, plotting pressure and using 15 contour values.

FIG. 1



TIME 110 μS



TIME 7 μS

FIG. 1 BASIC DENSITY CONTOUR PLOTS

FIG. 2

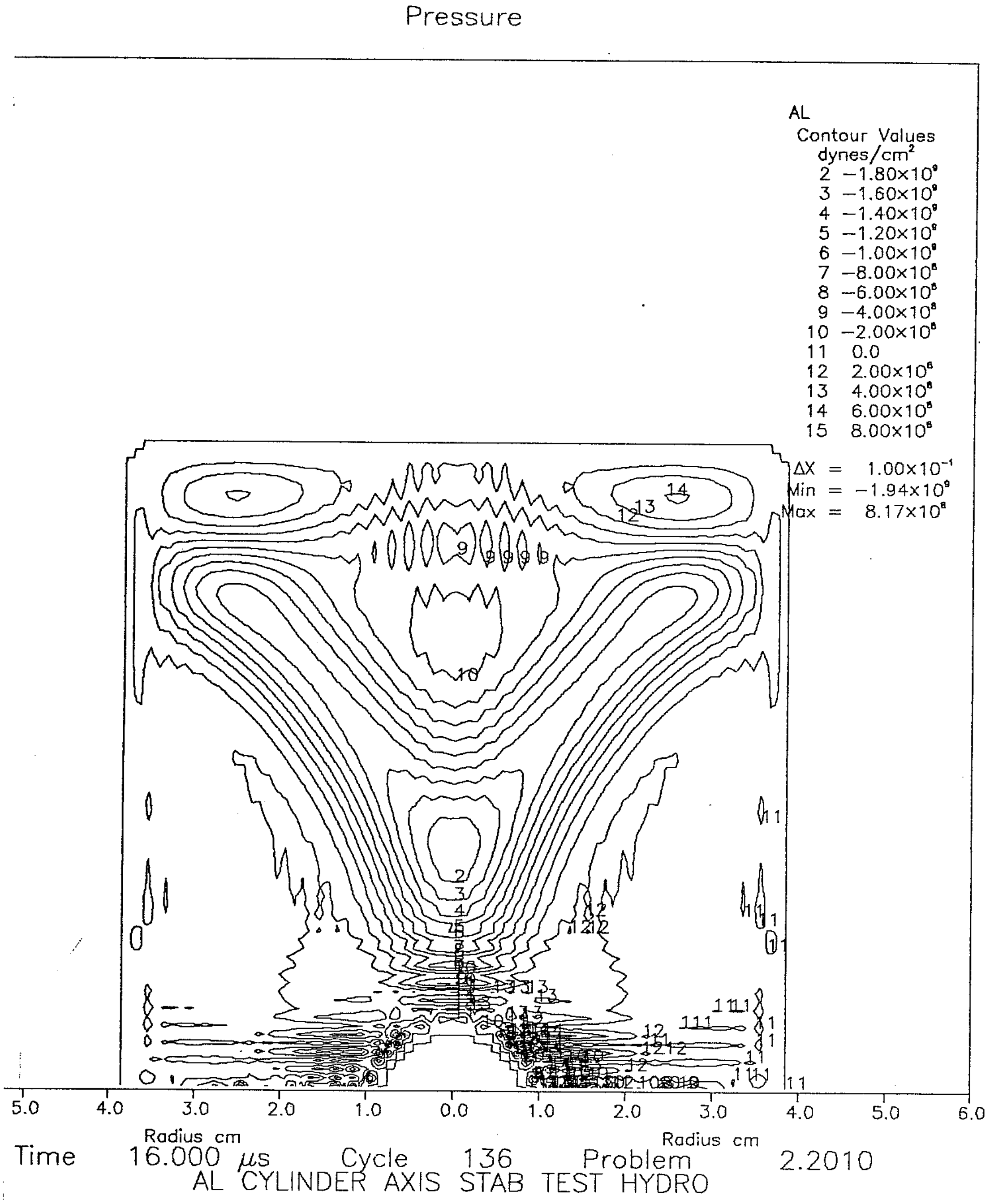


FIG. 2 STRESS WAVE CONTOURING

3.1 Carpet Plots

An alternative way of displaying data, particularly wave motion, is to plot the parameter value as a vertical height on a surface representative of the numerical mesh. This three dimensional surface or carpet plot (figure 3) provides far more visual information about the distribution of a parameter in a single plot. The plot in figure 3 displays the details of the pressure distribution in the detonation wave for the problem in figure 1. Notice this technique resolves the numerical structure in the wave front very well. It also has the added advantage of being viewable from any number of eye positions. This process becomes trivial on modern terminals with the facility to change the viewpoint locally. Although the use of carpet plots represent a powerful facility they do need careful interpretation, since in our experience, plotting/terminal display effects can sometimes lead to confusing "optical illusions".

In conjunction with contour plots, however, which supply "magnitude" information, carpet plots can maximise information recovery from a calculation dump.

On our CRAY 1S the generation of a single contour or carpet plot takes between 2-10 seconds cpu depending on the size of mesh and number of contour values. The real time acquisition of the data, however, can take significantly longer, especially with fragmented datasets on a fully loaded machine.

3.2 The Use Of Colour

Many of the problems of data representation, described above, can be easily overcome by the judicious use of colour. Various aspects of interest in a calculation can be enhanced by the use of specific colour ranges. We have found the raster scan display technique to be far superior to simply colouring an existing contour plot. In the simulation of stress wave propagation in a metal, for example, the representation of compressive stress states by red hues and tensile states by blue hues transforms plot interpretation. Equally a material in a calculation, for example the dished liner in figure 1, can be represented by a single hue, irrespective of density variations, to enhance its interface (it also has the advantage of hiding numerical errors!).

FIG. 3

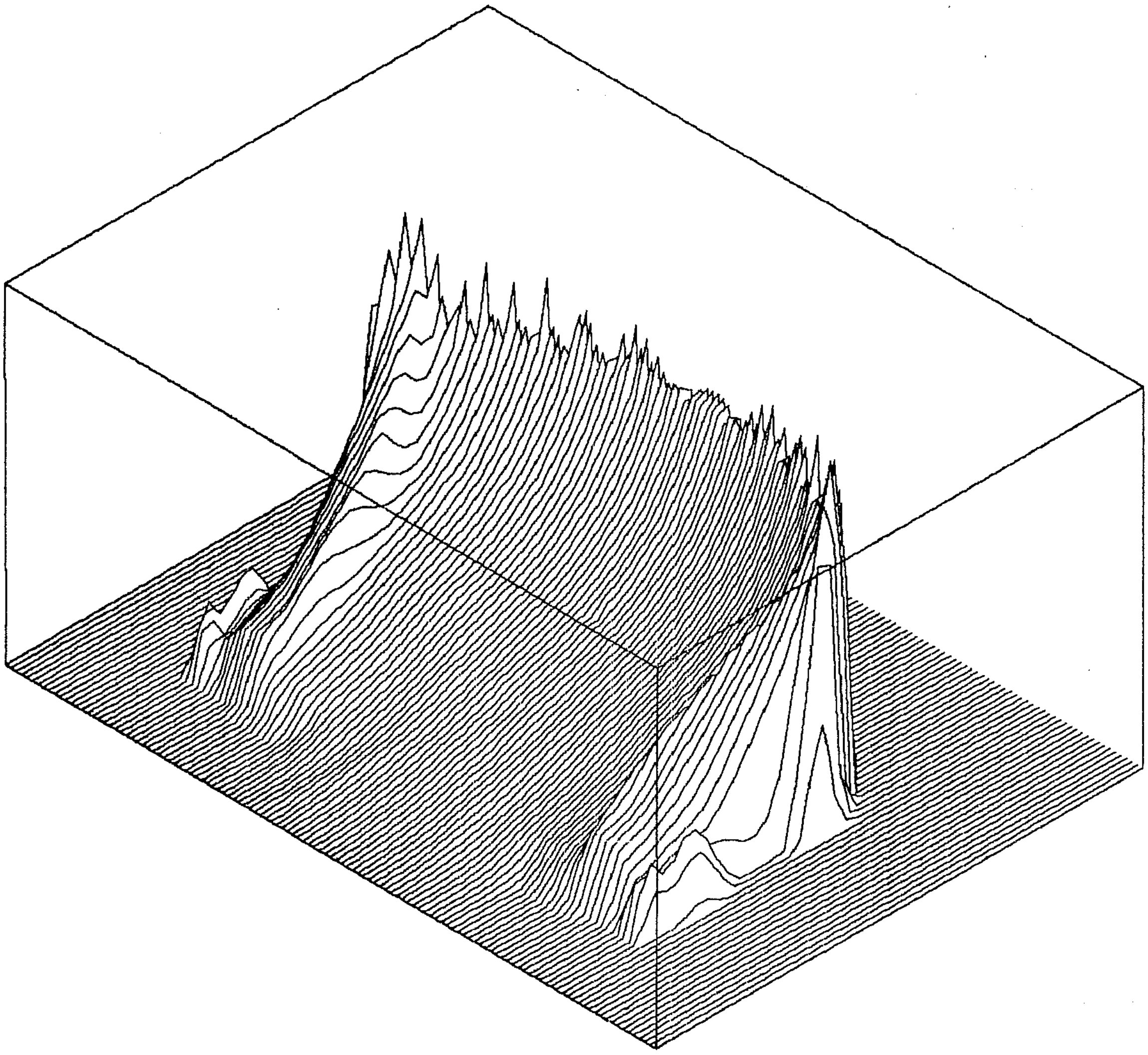


FIG. 3 PRESSURE CARPET PLOT

Alternatively, when concerned with algorithm development, colour values can be defined over a reduced parameter range to study small variations about a baseline value. This technique provides a powerful tool for identifying numerical instabilities, since with judicious choice of colours a classic "chequer board" pattern is produced in the region of the instability.

3.3 The Colour Graphics Package

The graphics package is designed to generate colour contour information for FE and FD codes using the raster-scan display technique, which utilises the terminal's capability of producing a television-like display. The package also contains device driver routines and higher level routines which are Calcomp compatable, although as yet do not conform to the Graphical Kernal System (GKS) standard. The package is written primarily for the Tektronix 4100 Series Terminals but can be modified for other terminals. There are, however, some features which make the package more efficient, but are not strictly necessary for other graphics terminals. In particular the Tektronix <runlength-write> command is included in this package to minimise the amount of data created for raster displays. The package has been tested on various computers (namely CRAY, VAX, IBM, CDC and SEL computers) at different sites to make it as machine independent as possible.

The software is written to obtain graphical data on a cell by cell basis from the computational mesh. It begins by relating the physical and computational cell co-ordinates in terms of pixel co-ordinates. A check is made to see if individual mesh cells are either square or rectangular, or if an expanded mesh has been used during calculation. The overall minimum length to represent any of the mesh cells is then computed and assigned one pixel. Thus the smallest square cell, assuming one has been found, is described by one pixel. This is the default. An input parameter however, can amend this as required. All remaining cell dimensions can now be scaled accordingly and assigned the appropriate number of pixels. This co-ordinate transformation is repeated every time a colour frame is requested as the pixel representation of a cell can alter during the calculation. In this way distortion due to variable mesh sizes is eliminated.

Having defined the mesh in terms of pixel co-ordinates, individual pixels are colour coded to represent the field variable that is of interest. A search is made of the calculated data for the cycle. The graphics package interrogates each cell for the required field variable, eg density, and assigns a colour index from either a default or a predetermined variable range. The colour indices are then arranged to represent the computational mesh on a row-by-row basis. This data is then compressed, for example, <runlength-write> encoded, and stored sequentially in a temproary file. This "metafile" is then disposed from the CRAY to the VAX where it is available for display purposes.

The form of the graphical output produced depends upon requirements specified in the input deck. Users may request a half plane colour plot of a specified cell variable. A mirror image option allows the user to reflect the plot about the axis of symmetry. There is also a split image option which allows a colour plot of two specified cell variables to be displayed side by side. For example a density plot is displayed on the left and a pressure plot on the right hand side of the screen. The user can define which portion of the computational mesh he wishes to plot, the default being the whole mesh. This option is particularly useful if the whole mesh has a pixel representation greater than the terminal resolution and cannot therefore be displayed on the screen. The compare option can access two different problem dumps and display their respective colour plots on the left and right hand of the screen. Where necessary, pixel replication is used with a scaling factor to permit magnification of the colour plot.

The graphics package makes every effort to minimise the amount of data generated for each colour frame. As an illustration of the techniques described above, consider two plots. The first plot consists of a series of short vectors drawing in imaginary terminal space of 4096 x 4096 addressable points. The second plot is an identical raster display plot making use of Tektronix <runlength-write> command and raster memory space of 640 x 480 pixels. As can be seen in Table 2, the raster display plot file contains the smallest amount of data per frame and can be displayed in a matter of seconds on the Tektronix screen. This is clearly a necessity when the link between the computer and terminal is via a communications network which may typically operate at data rates of 9,600 bits/second, or less.

DESCRIPTION OF PLOT FILE	DATAFILE SIZE	DISPLAY TIME	REDRAW TIME
General database containing (x,y) co-ordinants and pen up or pen down instructions	789,651 bytes		
Rasterised versatec plot	121,576 bytes		
Short vectors generated by Tektronix driver	72,409 bytes	100 seconds	20 seconds
Raster display using Tektronix runlength-write command	11,368 bytes	14 seconds	9 seconds
Raster display as before but with scaling factor = 2	18,946 bytes	22 seconds	12 seconds

TABLE 2: Comparison of Datafile Sizes

3.4 Interface Representation

As described earlier an eulerian computational cell, can contain more than one material, with average values for cell field variables, eg density and pressure. At interfaces with a large variable gradient, this averaging means that the colour assigned to the cell variable can be different from the colours assigned to any of the material species in the cell. Normally, this effect is visually minimised for fine mesh calculations and high resolution terminals. However, for coarse mesh calculations, or low resolution terminals, interfaces are incorrectly represented leading to poor quality colour images.

From a colour graphics point of view, the problem is to decide which colour index to assign to a mixed cell, regardless of a terminal screen's resolution or how many displayable colours are available. A number of algorithms have been studied to address this problem. The simplest solution attempts to assign meaningful colours to mixed cells. For example brown can be used to highlight the boundary areas around a material in the mesh. This approach is perfectly adequate when interface position is important.

A more complex algorithm, however, has been successfully developed to handle those situations where more than one pixel addressed a computational cell. The algorithm, instead of assigning the same colour value to each pixel in the cell, interpolates between cells and grades pixel hues accordingly.

3.5 Colouring Surface Plots

The use of colour in surface plots presents a number of different but interesting problems. In representing and viewing any surface, the surface is divided up into a number of panels. Each panel is shaded according to its orientation with respect to the eye position. Various methods have been developed in assigning a hue and intensity to each panel. We have found that whatever the shading algorithm used the most satisfactory visual results are obtained with a single hue varying in intensity.

There are three commonly used shading models.

The first, and simplest, uses a constant intensity for each panel. The intensity level assigned to the panel is defined in terms of the cosine of the angle between the panel normal and the eye position. Most basic 3D graphics terminals employ this shading algorithm.

The second, due to Gouraud, assigns an intensity level between adjacent panels. This algorithm is employed by many of the more sophisticated terminals such as the Tektronix 4129. (Foley 84)

The third method known as the metallic shading is due to Phong and assigns an intensity variation across a panel by interpolating normals between adjacent panels. (Foley 84)

In our experience, whilst the Gouraud method is computationally quicker it is not as visually impressive as the Phong approach.

The processing of each plot is carried out on the CRAY and down loaded to the terminal via the VAX front end machine. This provides us with our own versatile shading capability. We maintain a further flexibility by being able to process each plot in one of two ways; a raster plot, or a panel plot. Conversion of the panel plot to a raster plot minimises file size (40Kbytes for a 3D problem) and drawing time (90 sec at 9600 baud). The inability to change eye position locally, however, presents a major disadvantage. A new eye position requires reprocessing of the data.

Transmitting each individual panel, whilst generating considerably greater files (2Mbytes) and greater drawing times (20-30 minutes) has greater flexibility for additional local processing.

In our view having this flexible approach provides an extremely powerful facility.

4. ANIMATION TECHNIQUES

Whilst an extensive graphics capability can maximise information retrieval from a calculation dump, it does not address the problem of the limited number of dumps often associated with a calculation. Limiting the number of data dumps per calculation can result in important phenomena being completely omitted from the analysis. In addition, rapid data processing will only give an overview of the calculation. For example, the formation of a mach wave due to intersection of two oblique shock waves occurs within a few time step intervals during the calculation. This event will be only observed if the user has anticipated its formation and duration and has produced data dumps at the required times. If specific events are to be observed, it may be necessary to re-run parts of the calculation or fragment the calculation in order to complete a full analysis.

At RARDE we have extended our raster graphics package to overcome these problems. The user can request colour "snap shots" of events, at every step interval if necessary, without generating too large a data file, and review the calculation more efficiently as the calculation is running. He can then rapidly assess the importance of transient pheonomena in a calculation, investigate numerical stability or the effect of new algorithms, or decide if parts of the calculation need to be re-run for an in-depth analysis.

The availability of this graphics information throughout the calculation has introduced the possibility of generating results in the form of animated sequences. The Theoretical Warhead Design Section within RARDE developed in 1984 a system for automatically recording computer generated colour frames directly onto a video medium. Reintroducing time back into a calculation has a number of important implications for hydrocode studies.

It identifies the timescale over which different physical processes are dominant together with their relative importance.

By identifying these processes a calculation can be rerun with increased data collection over a narrow time period. All the facilities available in post processing, eg split plots etc, are available for animation. Thus pressure and density changes can be followed together through a calculation. Finally a numerical instability can be localised to a given region of the mesh and/or its propagation through the mesh followed in time.

4.1 Video System Description

The system is currently based around the Tektronix 4113A terminal, although research is going on into applications with modern 3D terminals (Tektronix 4129, Iris).

Of the terminals on the market in 1983/4, the Tektronix 4113A best suited these requirements. This terminal was therefore purchased as a working terminal with the peripheral ports providing a means of linking a local printer and graphics plotter. In addition the terminal was also configured with an external video board which provided a 25Hz interlaced RGB video output signal. This video option allows users to display frames on a remote TV monitor.

Two options were available at that time for recording colour graphics output. The first involved piping the RGB signal into a digital film recorder (matrix camera) for producing slides, prints and cine film. The alternative was to store the colour frames onto video disks and then transfer them onto video tape.

The matrix camera proved to be a suitable means of obtaining slides or prints of computer generated colour plots. Superimposing red, green and then blue frames onto a colour negative increased the colour range from 16 to 256 colours. However, the making of cine films did not prove attractive. Exposures were permanent. Mistakes were not obvious until after the film had been developed. The process was time consuming. Finally the system was not portable as matrix camera settings had be re-adjusted every time it was attached to another graphics terminal with a similar RGB signal outputs.

The video recording system developed at RARDE proved to be an effective and inexpensive way of animating graphical output. As this system has been automated, a sequence of colour frames can be stored automatically and unattended onto magnetic video disks in under an hour. This sequence can then be previewed for display continuity and colour content. The end product, which is transferred onto video tape, is immediately available for presentation at minimal cost. Finally, the system is fairly portable.

The equipment used in this sytem consists of two Eigen J20 Magnetic Video Disk Recorder Players (VDR), a Disk Controller, Time Base Corrector (TBC), Cox Encoder, Video Tape Recorder (VTR), Video Typewriter consisting of a keyboard unit and control unit, and a Video Monitor. The system is illustrated in figure 4 based around the Tektronix 4113. Other terminals, however, having the correct video characteristics could likewise be interfaced with this equipment.

In operation the external video board in the Tektronix 4113A terminal provides a continuous 25Hz RGB vidoe output signal. This signal is passed through the Cox Encoder where it is converted to a standard composite PAL signal. The signal then loops through the TBC before it is recorded in analogue form on the master and slave VDRs. If a permanent record is made of this signal on video tape, the signal is taken from the VDRs and once again looped through the TBC to remove signal and timing errors before being stored on the VTR. There is, however, a limitation of the digital-to-analogue system. If the Cox Encoder, the TBC and the VDR were connected to a terminal allowing 256 displayable colours, the number of displayable colours will have been reduced when viewed on the VTR because of the increase in signal-to-noise ratio.

FIG. 4

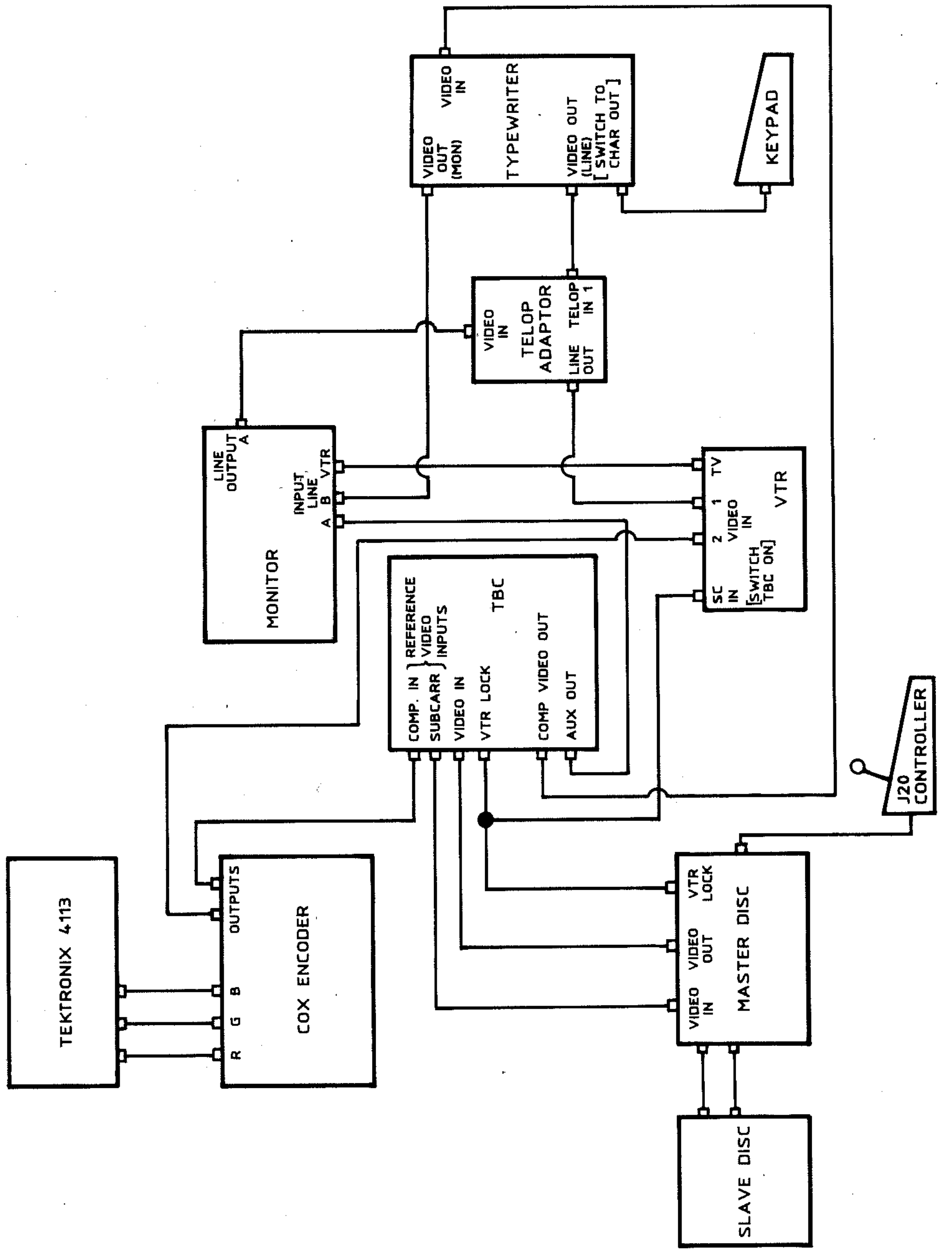


FIG. 4 VIDEO EQUIPMENT CIRCUIT

A VAX based program controls the display and storage of each frame from the calculation onto the video disks. When all the colour frames have been stored on the VDR they can be transferred to UMatic video tape. This video tape medium has been found to be very satisfactory for our use in assessment work as well as presentation. If necessary the video sequence can be transferred to broadcast quality video tape or 16mm film when required for external presentation at very little additional cost.

There are several advantages in using the RARDE system. Installation, maintenance and running costs are relatively low. The equipment is easily operated as the recording sequence is fully automated. It is flexible since mistakes can be rectified with little effort and alternative recording equipment can be connected to the VDRs. Up to 1000 tracks (ie 500 TV frames) can be stored onto the VDR permitting 20 seconds of video display at normal speed. Finally, any of the stored tracks can be transferred onto the VTR at variable speeds producing slow motion recordings, a technique that significantly aids the understanding of many calculations. In the future optical disks could be connected to the sytem thus providing a permanent high quality archive facility.

4.2 Future Research

Research is already underway to investigate methods of adapting our animation techniques for the modern 3D graphics terminals and workstations.

The limited range of hues available on the Tektronix 4113restricts the visual effectiveness of our surface plots. Nevertheless early animations of surface plots have been encouraging. We have commenced investigations into animation techniques using the Tektronix 4129 terminal with its 256 colour option. Unfortunately the lack of a video output board has precluded rigorous tests to isolate the problems associated with the increased signal to noise ratio generated by 256 displayed colours. Currently we are using a video camera to interface our video equipment with the 4129, and whilst this arrangement is less than satisfactory and time consuming the results are quite exciting.

In the near future we intend investigating applications using an Iris workstation.

5. SURFACE GRAPHICS IN TWO AND THREE DIMENSIONS

The immense amount of data generated in an eulerian hydrocode calculation has already been highlighted. The graphics techniques described in this paper have all been designed to effectively display this data in a form that is easy to assimilate. In particular the animation technique has proved an extremely powerful and wide ranging capability.

The majority of our research is geared to "metal bashing", whether forming high velocity penetrators with explosive, or studying their impact against a target. There is therefore a requirement to generate realistic pictures of solid objects from our calculations. This is particularly true for three dimensional simulations, since the graphics techniques normally available can only operate on a vertical or horizontal plane. To understand for example how a cube behaves as it impacts and penetrates a thin plate only material interface plots in vertical or horizontal planes is difficult and rather tedious. Very often information is lost because the wrong planes were plotted. Being able to visualise the object in a solid form, therefore, is of immense benefit. The generation of the material interface in a solid form, in a lagrange calculation is a relatively straightforward task since the data on the position and form of the interface is automatically defined by the mesh. In an eulerian calculation all that is known is that a material interface lies somewhere in a cell.

To achieve this objective in eulerian codes, therefore, two codes were developed; BOOM for two dimensional axisymmetric calculations and MAST for three dimensional calculations. Both produce a plot that is a two dimensional projection of a three dimensional object rendered using illumination and shading techniques.

5.2 Boom

Boom is the solid surface program designed to produce pictures of solid objects defined by 2D axisymmetric calculation data. It introduces the third dimension by rotating a 2D dataset about its axis of symmetry The resulting solid surface can then be manipulated and displayed as required. Boom has two main advantages over conventional 3D solid modelling programs; its low memory requirements, and its very high execution speeds, typically 2 seconds on a CRAY 1S.

5.3 Boom Data Input

The basic input data required is a 2D array of density values, arranged on a regular grid, usually obtained from a Hull hydrocode calculation dump. A contour is defined within the eulerian data, thus fitting an outline around the specified material(s) present in the mesh. Although we use density contours to describe the outline of solid objects, any other parameter, eg pressure could be contoured.

Alternatively the user can supply a set of X, Y coordinates that defines a given contour and use Boom as a simple solid modeller.

5.4 Boom Surface Generator

The contour from the 2D dataset is output as a series of points in the plane of the mesh. To restore the third dimension to the problem, it is necessary to rotate the contour about its axis of symmetry. This is done by applying a transformation to each point in the contour. The rotation is carried out in a series of discrete steps through a fixed angle. When the contour has been rotated through 360°, the points lie on the surface of the solid object represented by the 2D axisymmetric data held within the program array.

5.5 Display

The resulting set of three dimensional points can be displayed in a variety of ways.

The simplest is as a wireframe representation of the surface, with or without hidden line removal. This is built up by joining adjacent points to form a grid or mesh around the surface of the object. Realism can be enhanced by adding perspective and removing hidden lines from the picture.

Further enhanced realism can be achieved by adding illumination and surface shading to the object. The quadrilateral panels representing the surface of the object are formed from the contouring points.

These panels can then be shaded using one of the shading techniques described previously.

Boom therefore provides a convenient way of quickly recreating the material interfaces within a 2D axisymmetric simulation in a recognisable form.

Recreating and visualising material interfaces in this way improves our understanding of the physics involved in penetrator formation and its subsequent behaviour in target interactions.

Some examples of Boom output are illustrated in figures 5-7. Figure 5 displays the formation of the iron dished liner in figure 1 into a projectile travelling at 2000m/s.

Figure 6 illustrates the collapse of a conical copper liner to produce a hypervelocity jet whose tip travels at 8000m/s.

Figure 7 illustrates the use of Boom with a user defined set of X, Y coordinates that contour a notched tensile test specimen used in our material property tests.

5.6. Mast

When faced with a true 3D eulerian calculation, where no axes of symmetry exist, and simple line contouring of an eulerian dataset cannot be used, new techniques are required. Mast was designed to extend the techniques where unlike lagrangian data, the surface is defined explicitly. To produce realistic pictures of the objects interacting within the simulation, it is first necessary to extract the outline of the solids within the problem. This effectively reduces to the task of fitting contour surfaces through the 3D dataset, in a similar fashion to the contouring of 2D data by fitting lines through the mesh.

5.7 Mast Plotting Data

In order to display the outer surfaces of the solid objects being modelled by the simulation, it is necessary to contour the data to extract the air-solid interfaces bounding the objects within the problem. As we only require density for contouring, the dataset is processed to extract only the density data, reducing the amount of

FIG. 5

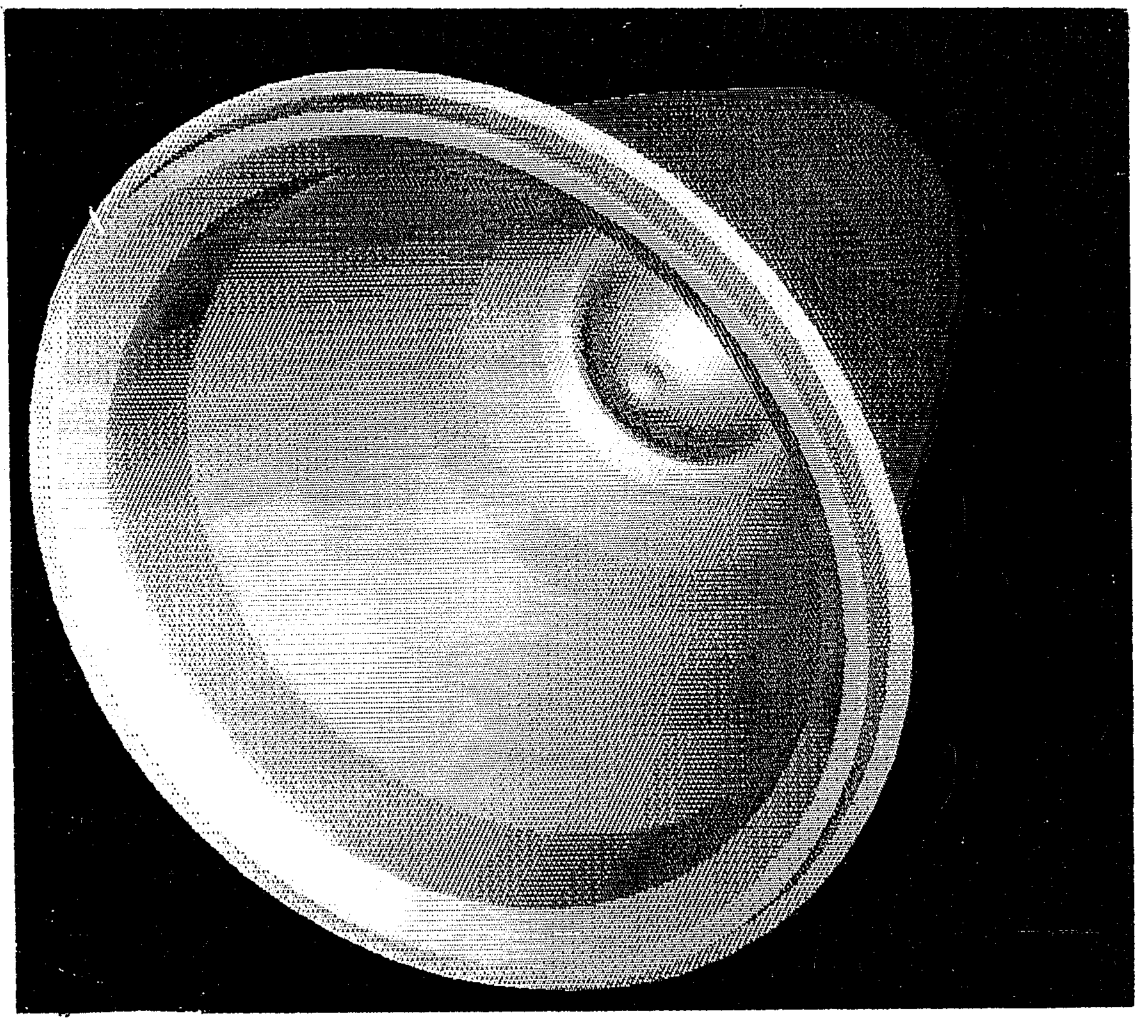


FIG. 5 BOOM : PROJECTILE FORMATION

FIG .6

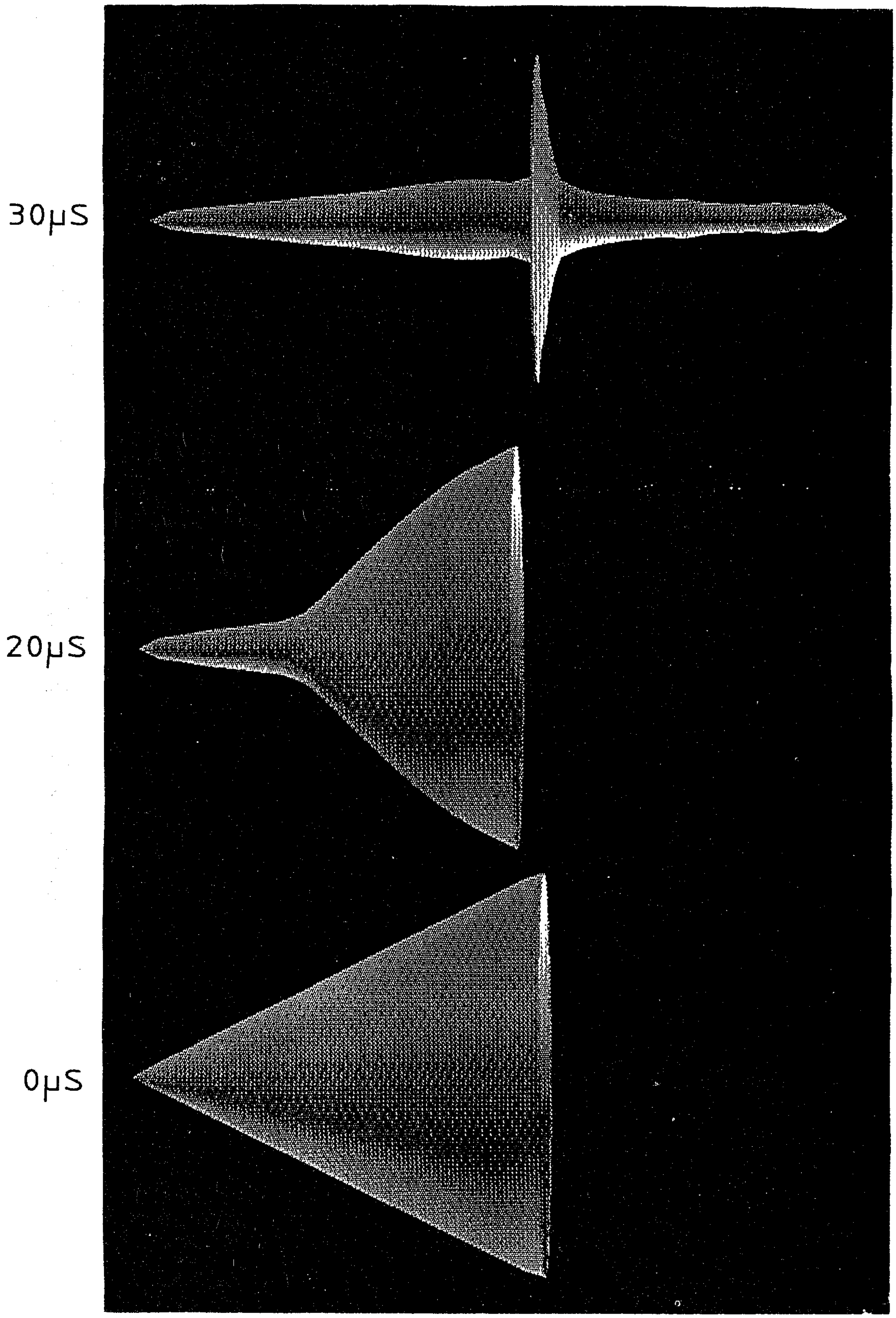


FIG. 6 BOOM : CONICAL LINER COLLAPSE

FIG. 7

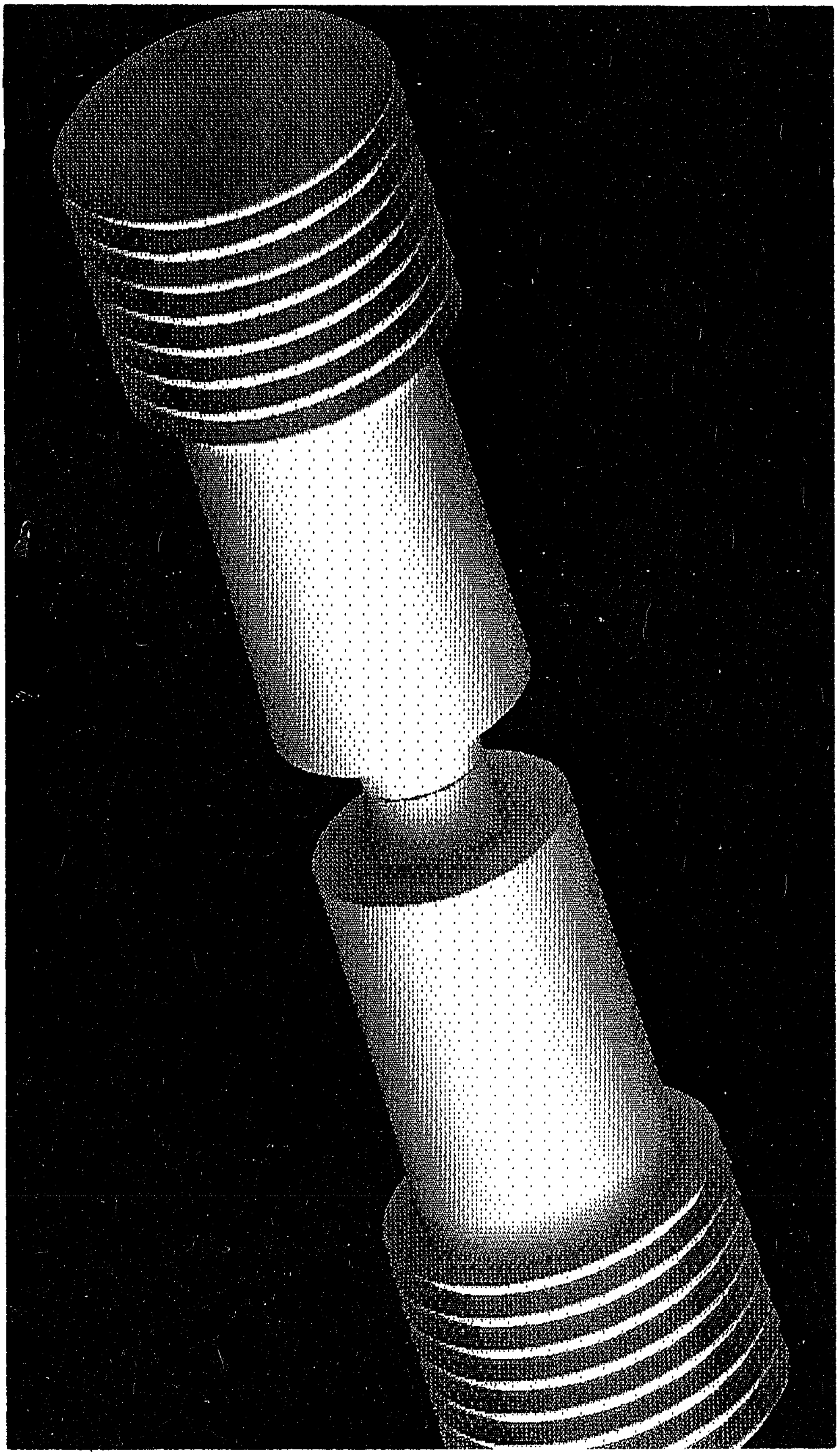


FIG. 7 BOOM: USER DEFINED INPUT

data to manageable proprotions. There is no reason, however, why any other type of data could not be contoured, producing pressure surfaces for example. The only restriction placed on data is that it must form a regular mesh, random sampling points are not suitable. On our machine, the CRAY 1S, i/o times are the main problem, so all data is held within program arrays.

5.8 Mast Contouring

Mast contours the eulerian dataset cell by cell, taking cuboids of data from the mesh. Each cuboid will have eight density values associated with it, one at each vertex.

The program then fits a surface through the cuboid corresponding to the contouring value selected in the input. Simple checks are made to determine whether a contour surface passes through the cuboid. If all vertices are higher (or lower) in value then the contouring value, then a surface cannot pass through the cuboid. It can then be discarded and contouring moves to the next cuboid.

If a cuboid containing part of the contour surface is found, then the program attempts to fit the surface through it. Although the contour may be a complex curved surface, the program simplifies this to a single n-sided panel within the cuboid. This reduces computing time considerably and, for large meshes, produces little deterioration in the final results.

The contour surface within the cuboid is determined by finding its intersections with each face of the cuboid. Each face is contoured separately to determine the intersections to be joined. They thus form one or more panels representing the contouring surfaces passing through the cuboid. The advantage of this method is that adjacent cuboids share a common face, panels produced will share common edges. Thus the contour surface can be built up as a series of panels which will automatically join together.

What the contouring procedure does is convert an eulerian dataset, with no defined boundaries, into a lagrangian one, where boundaries are defined explicitly.

5.9 Mast Display

Once all cuboids have been contoured, and the contouring surface obtained, the surface can be displayed.

For our application, the surface produced represents the outlines of the solid objects being modelled. We therefore display it as an illuminated solid object, using the shading models described previously. Shading can either be done entirely within the program, or data can be sent to suitable 3D terminals to enable them to perform surface shading. If program shading is used, more realistic shading models may be employed, and the plot files are more compact. The advantage of terminal shading is that it allows the objects to be rotated and shaded locally, without having to rerun the plot program.

For added realism, several such shaded object pictures can be produced, at regular simulation time intervals and animated as described previously. We can thus display an animated sequence of pictures that accurately represents the interactions between objects that we have simulated.

An example of the use of Mast in a three dimensional penetration study is illustrated in figures 8-10. Figure 8 shows the initial rod-plate orientation, and figures 9 and 10 the penetration process at 15 and 20 μseconds respectively.

6. SUMMARY

This paper has attempted to illustrate the way we have developed our graphics capabilities within the Warheads Division RARDE in support of eulerian hydrocode simulations of problems in detonics and penetration mechanics. Our evolved philosophy on graphics is best summed up by the word "flexibility".

We believe that, particularly in a research environment, it is essential to maintain and develop a single in-house multi-purpose graphics package. The effort in supporting many distinct packages can rapidly outstrip available resources.

Invariably, we have found that commercial graphics packages do not offer the facilities we require, expecially if we wish to try a new

FIG. 8

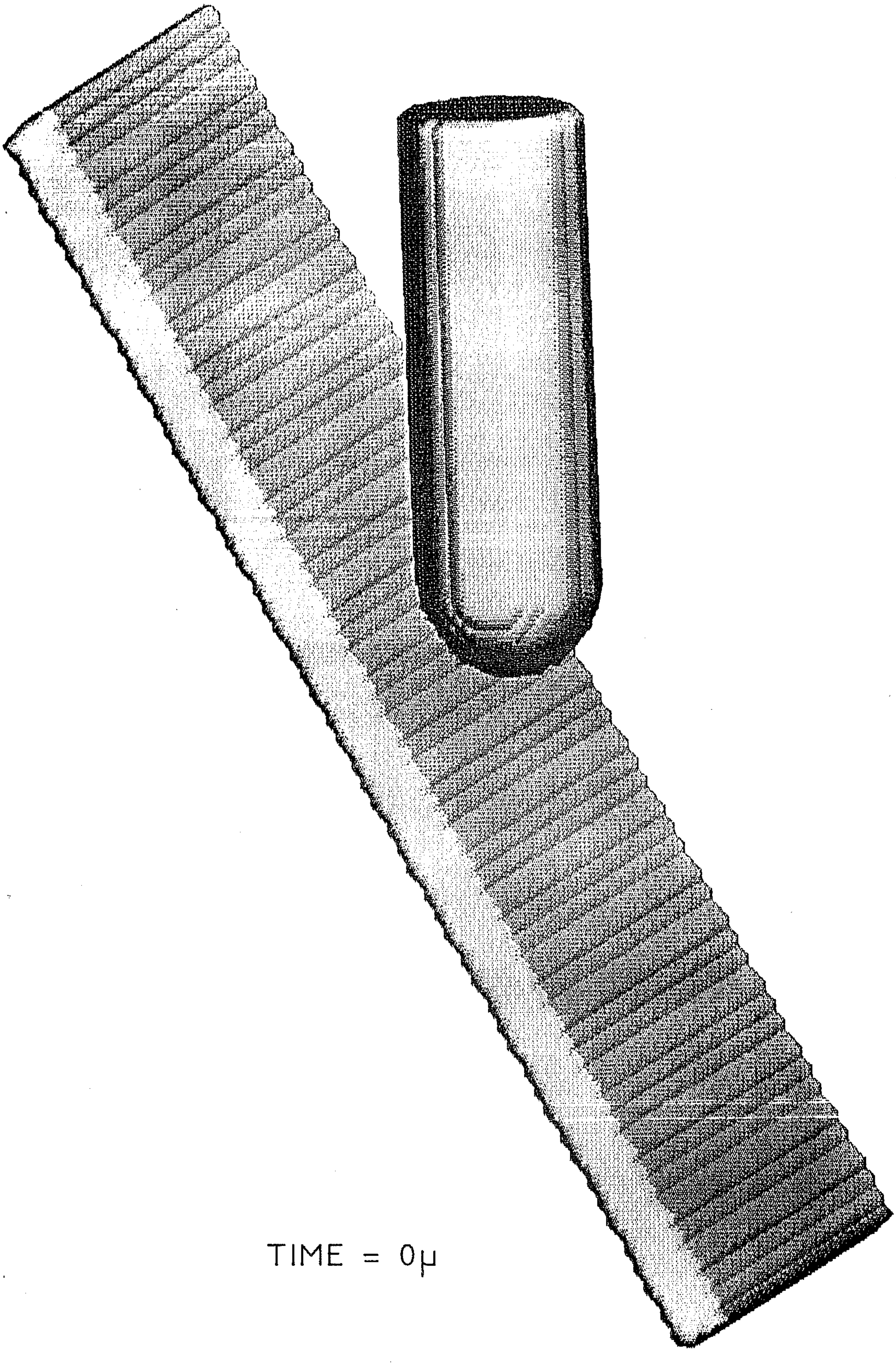


FIG. 8 MAST : ROD IMPACT AT 60°

FIG. 9

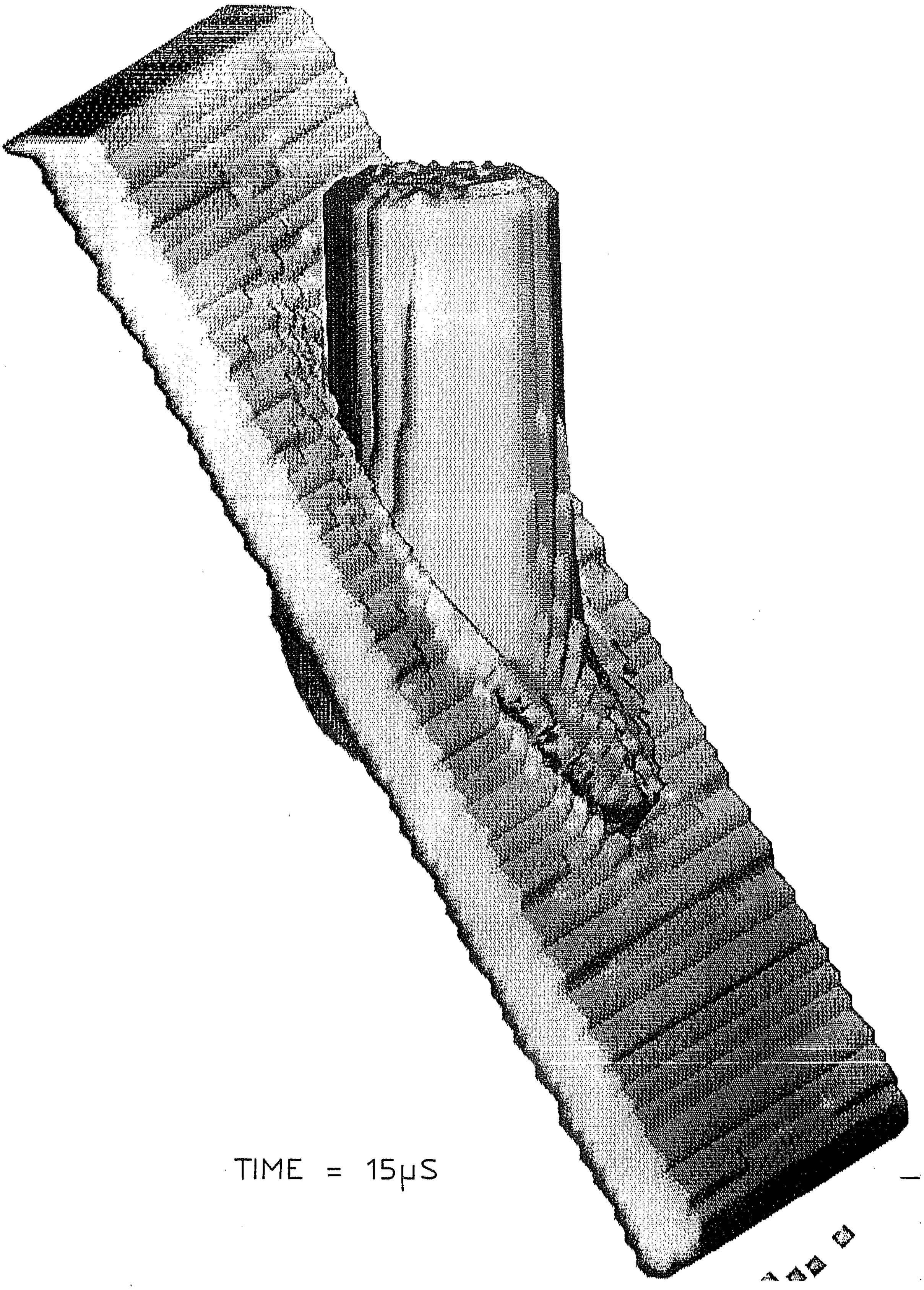


FIG. 9 MAST: ROD IMPACT AT 60°

FIG. 10

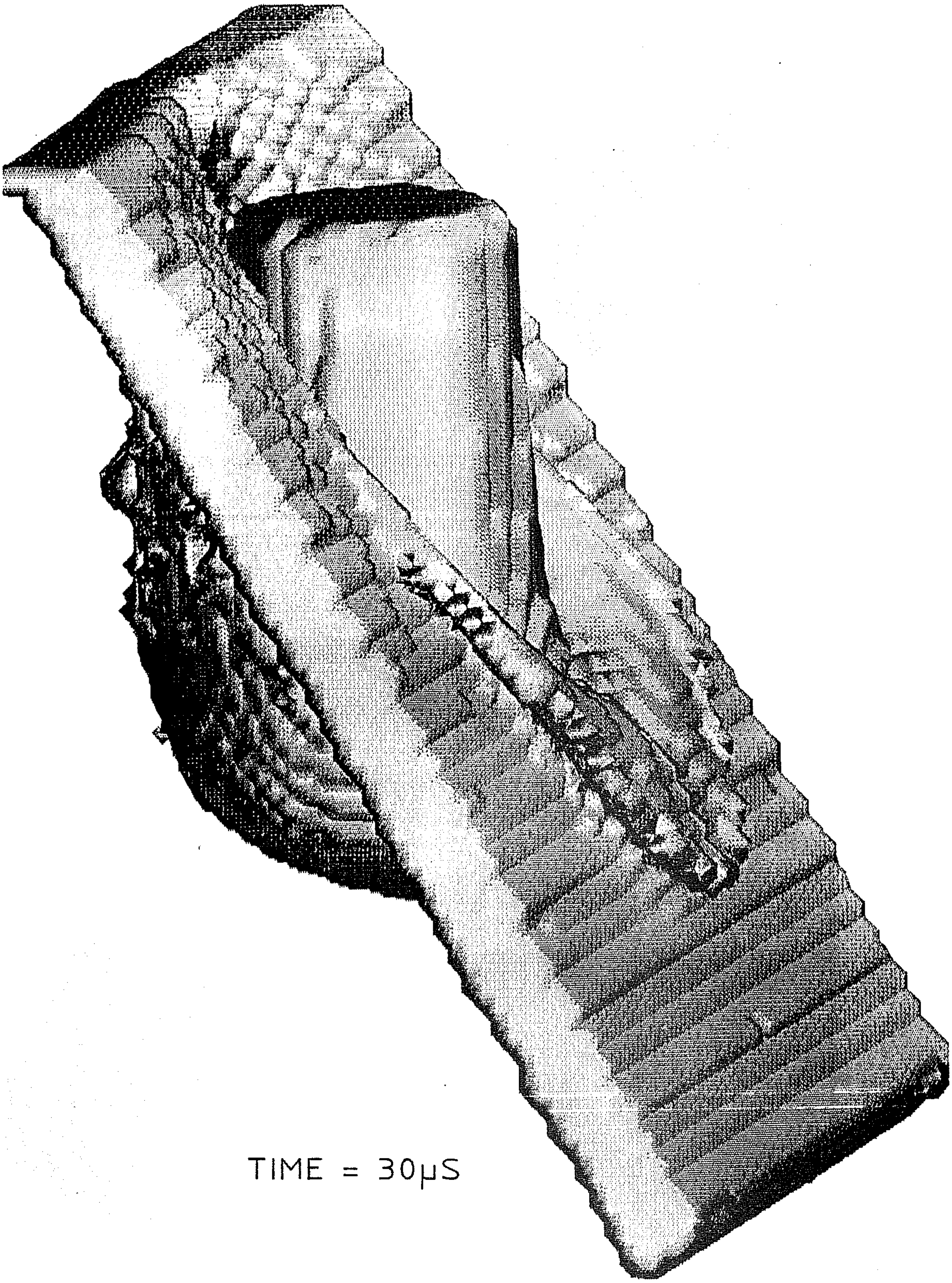


FIG. 10 MAST : ROD IMPACT AT 60°

idea or algorithm, or do "something clever" with the terminal.

In our experience it is far more cost effective to maintain a careful balance between the amount of processing performed centrally and that done locally on smart 3D terminals and workstations. The former allows far more flexibility in terms of graphics techniques, data transfer and cpu time. The latter, of course, allows local analysis of a subset of the data from a selection of viewpoints. Judicious use of both techniques maximises resource efficiency and job throughput, although the host-terminal communications link is clearly a limiting factor. As workstation performance/price ratios rise and more local processing becomes available this balance will obviously shift. Nevertheless a large proportion of our graphics work will in the future not require such sophistication, and will be most cost effectively supported by "dumb terminals".

The techniques described here obviously lend themselves to an interactive graphics environment, although most of our current graphics work is post processed, due to computer resource limitations. Once we replace our single processor CRAY 1S with an XMP or CRAY 2 we will be able to use a second processor to interrogate a calculations whilst it is running. Communication links will again be the limiting factor.

Whilst an interactive graphics environment has some benefits to us in improving job throughput we view with some concern the current trend in the USA to replace all terminals with workstations. This total reliance on interactive graphics by inexperienced users runs the risk of ignoring numerical effects and instabilities and believing in what they see on the screen. In our opinion worrying about the numbers is in many ways more important than looking at the pretty pictures.

Clearly, however, the rapid advances in computer/terminal technologies will make the next five years in computational physics extremely exciting.

7. REFERENCES

Anderson, C., 1986: Hypervelocity Impact, Proceedings of the 1986 Symposium. Pergamon Press.

Cullis, I., 1986: 9th International Symposium on Ballistics, pages 2351 to 2363 and 2285 to 2291.

Foley; Van Dam, 1984: Fundamentals of Computer Interactive Graphics. Adison Wesley.

Zukas, J., 1982: Impact Dynamics: Wiley