

KB2

**Field Precision**

Copyright 2001

PO Box 13595, Albuquerque, New Mexico 87192  
Telephone: 505-296-6689      FAX: 505-294-0222  
E Mail: [techinfo@fieldp.com](mailto:techinfo@fieldp.com)      Internet: [www.fieldp.com](http://www.fieldp.com)

## **Table of contents**

**Section 1. Introduction**

**Section 2. Running KB2**

**Section 3. Mesh generation techniques**

**Section 4. Structure of the KB2 command script (\*)**

**Section 5. Program control commands**

**Section 6. Applied pressure waveforms**

**Section 7. Defining standard materials**

**Section 8. Defining explosive materials**

**Section 9. Regions - element properties**

**Section 10. Regions - node properties**

**Section 11. Diagnostic commands**

**Section 12. Features of KB2V**

**Section 13. KB program launcher**

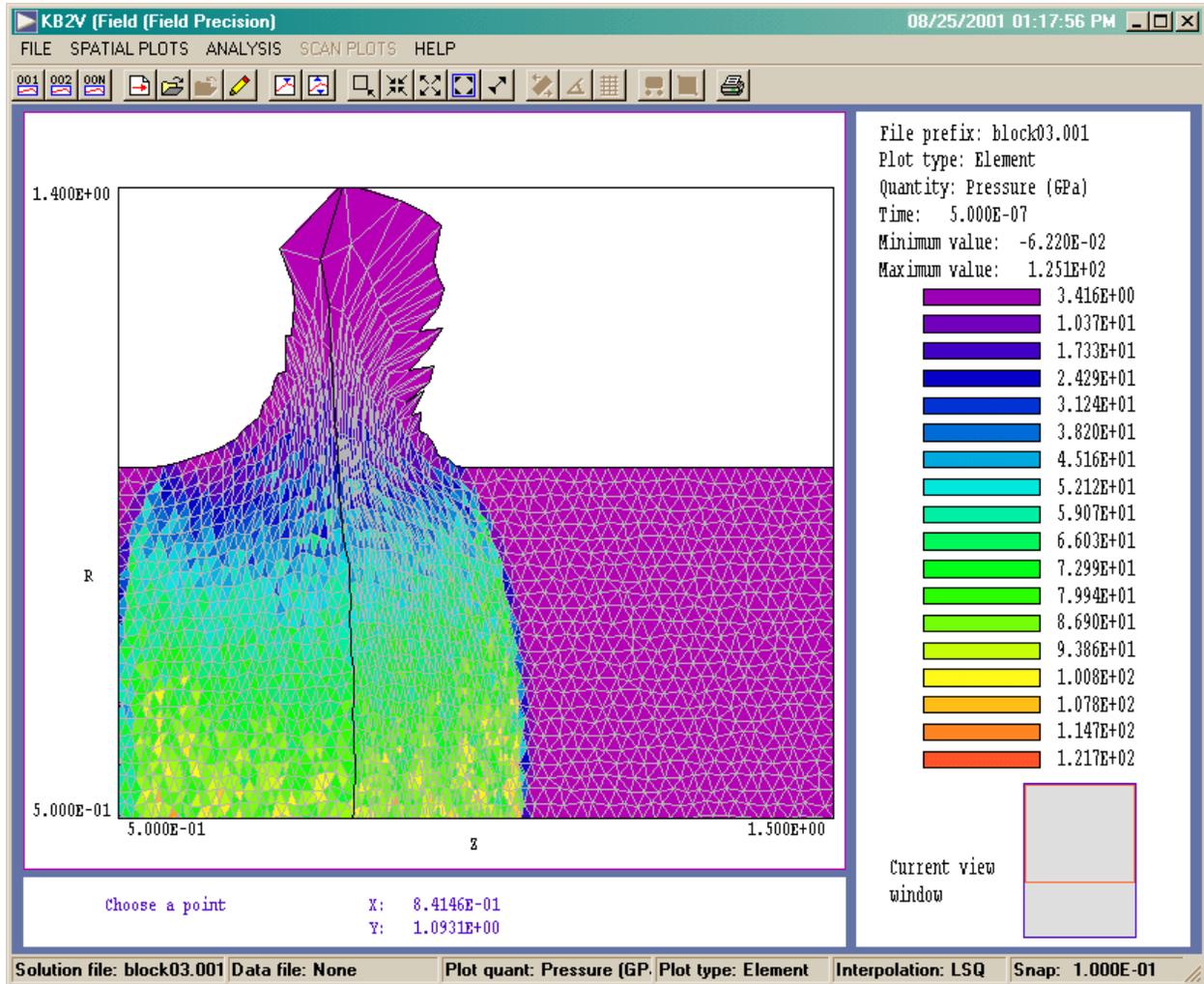


Figure 1. KB2V postprocessor screen shot - BLOCK03 example

## Section 1. Introduction

**KB2** is a versatile software system for simulations of shock hydrodynamics. The program handles two-dimensional cylindrical or planar geometries through a unique finite-element method that is well-suited to systems that undergo large changes in volume. This manual covers procedures to set up **KB2** runs. The program employs the **Mesh** program, the standard mesh generator for the Field Precision **TriComp** series of two-dimensional finite-element codes. Once the mesh for the initial system geometry is created, the commands to control **KB2** runs are straightforward. The mathematical foundation of the code is described in the manual *Physics of the KB Codes*. The software package consists of the following programs:

**Mesh**

Versatile utility to create conformal triangular meshes from user-specified geometries.

**KB2**

Main simulation program controlled by an input script.

**KB2V**

Interactive graphical post-processor for data files (spatial variations of quantities at specified times)

**Probe**

Standard Field Precision plotting and oscilloscope utility, used to inspect and to plot **KB2** probe files.

**Probe** is described in a separate manual. Table 1 summarizes physical units used in **KB2** and **KB2V**.

The following section summarizes the organization of **KB2** runs. A detailed description of an example is included to familiarize you with the format of input data and the relationship of components in the software system. The two required input files are:

# output file from Mesh describing the initial geometry of the system, and

# **KB2** script that provides physical information for the simulation.

Section 3 covers special techniques used in **Mesh** to create conformal triangular meshes for hydrodynamic simulations. The material supplements the manuals supplied with the **Mesh** and **XLate** programs. Section 4 covers the structure and syntax conventions of the **KB2** control script. The next seven sections cover the set of commands and parameters that may appear in the control script. Section 5 reviews commands for general program control. Controllable quantities include the time-step to advance the hydrodynamic equations, the total run time, and the interpolation method used for equation-of-state calculations. Section 6 covers methods to define applied pressure waveforms. Pressures applied on the system boundaries are one method to initiate shock hydrodynamic processes in **KB2**. The other two methods are:

# assignment of initial velocities to regions of the solution space,  
and

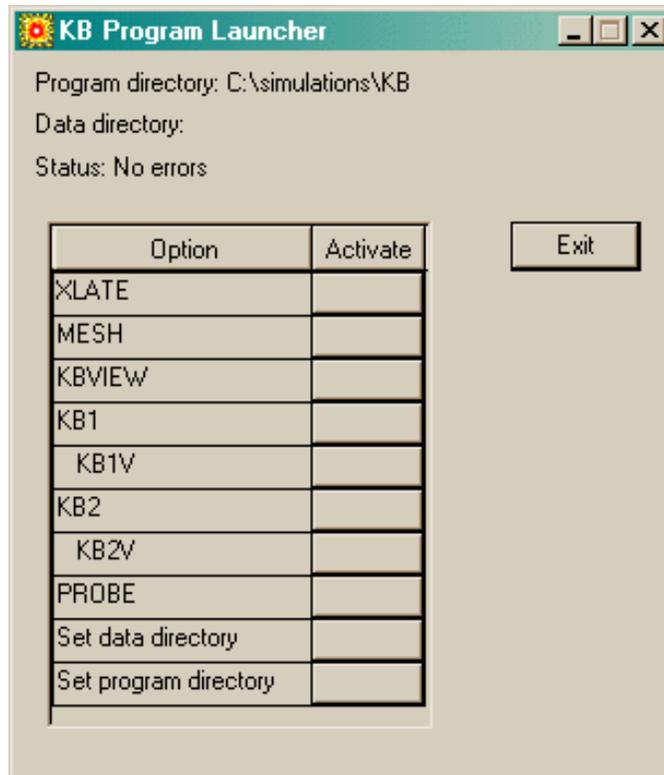
# detonation of explosive materials.

Section 7 summarizes commands to define the physical properties of standard materials while Section 8 covers explosives. Section 9 covers commands to set properties of properties of elements in *FILLED* regions. Parameters include the initial hydrodynamic conditions and the artificial viscosity coefficient. Section 10 describes commands to set node properties such as the initial velocity in both *FILLED* and *OPEN* regions. Special properties can be assigned to *OPEN* regions to define symmetry boundaries. Commands to control run diagnostics are described in Section 11. Section 13 describes features of **KB2V**. Finally, Section 14 covers the **KB** program launcher.

<b>Table 1.1. Physical units for the KB2 programs</b>		
Quantity	<b>KB2:</b> internal operations, data and history files	<b>KB2V and PROBE</b>
Density	kg/m <sup>3</sup>	gm/cm <sup>3</sup>
Position	m	Set by DUnit
Pressure	Pa	GPa
Temperature	°K	°K
Internal energy	J/kg	MJ/kg
Velocity	m/s	km/s

## Section 2. Running KB2

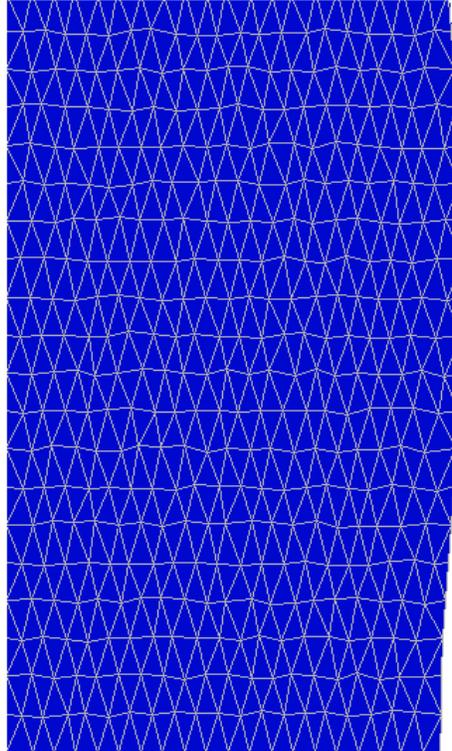
As an introduction we shall walk through the steps of a hydrodynamic solution. It is assumed that the executable files `KB.EXE`, `MESH.EXE`, `KB2.EXE` and `KB2V.EXE` are in the directory `\KB`. We need a temporary location to store input and output files, so create a directory `\KB\BUFFER`. Move the files `RT_INSTAB.MIN`, `RT_INSTAB.KIN`, `ALUM3715.KBT` and `STEPFUNC.DRV` to the temporary directory.



**Figure 2.** KB Program launcher

Run the program KB . EXE to display the screen shown in Fig. 2. The program directory should be automatically set to \KB. We need to set the data directory so that launched programs know where to look for input files. Choose the *Set data directory command*. In the dialog, move to the directory \KB\BUFFER and click *OK*. Section 14 gives more details on the operation of KB.

Figure 3 shows the initial simulation geometry for the RT\_INSTAB example. A 1 mm thick aluminum plate is accelerated by a zero-mass fluid on the right-hand side with peak pressure 5 GPa. The applied pressure follows a step-function variation with a risetime of 0.1  $\mu$ s. The plate surface has an initial perturbation on the upstream side such that the plate thickness varies from 1.00 mm to 1.05 mm. The sinusoidal variation has a wavelength of 2.5 mm in the vertical direction. The simulation represents 10 mm of a plate that extends infinitely in the vertical direction. Special symmetry boundaries are defined at the top and bottom. Figure 3 shows a portion of the plate along with the outlines of mesh elements from  $y = 0.0$  to 2.0 mm.



**Figure 3.** Mesh for the initial geometry of the example RT\_INSTAB.

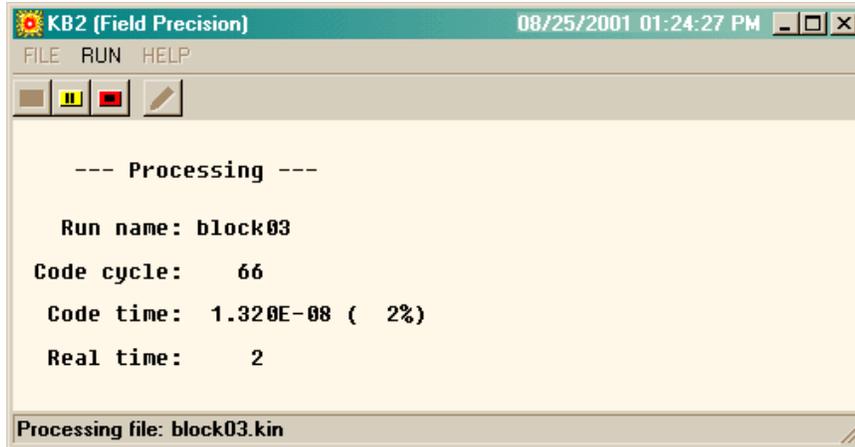
Run **Mesh** by clicking on the entry of the **KB** program launcher. Click on the *Load script (MIN)* command of the *File* menu. If all files are in the correct location, the choice in the dialog should include RT\_INSTAB.MIN.. Pick the file and click *OK*. Next, click on the *Process* command. The program analyzes specifications in the script to set up the geometry of Fig. 3. Click a mouse button to proceed. Finally, pick the command *Save file (MOU)* in the *File* menu. At this point, you have created the file RT\_INSTAB.MOU, one of the required inputs for the **KB2** run. At this point you can experiment with some of the plotting capabilities of **Mesh**.

Before leaving **Mesh**, it is useful to inspect the input script RT\_INSTAB.MIN. In the *File* menu, click on the command *Edit current script (MIN)*. A shortened version of the file is shown below. Two lines in the *GLOBAL* section are of interest

TriType Glass 0.10  
Smooth 0

### Mesh input file RT\_INSTAB.MIN

```
GLOBAL
  XMesh
    0.00 1.10 0.05
  End
  YMesh
    0.00 10.00 0.100
  End
  TriType Glass 0.10
  Smooth 0
END
REGION FILL 1
* Aluminum flyer plate
  L 0.00 0.00 1.00 0.00
  L 1.00000 0.00000 1.00079 0.10000
  L 1.00079 0.10000 1.00309 0.20000
  ...
  L 1.00309 9.80000 1.00079 9.90000
  L 1.00079 9.90000 1.00000 10.00000
  L 1.00 10.00 0.00 10.00
  L 0.00 10.00 0.00 0.00
END
REGION 2
* Slide boundary: top
  L 0.00 10.00 1.00 10.00
END
REGION 3
* Slide boundary: bottom
  L 0.00 0.00 1.00 0.00
END
REGION 4
* Pressure boundary: right-hand side
  L 1.00000 0.00000 1.00079 0.10000
  L 1.00079 0.10000 1.00309 0.20000
  ...
  L 1.00309 9.80000 1.00079 9.90000
  L 1.00079 9.90000 1.00000 10.00000
END
REGION 5
* Pressure and slide point: top
  P 1.00 10.00
END
REGION 6
* Pressure and slide point: bottom
  P 1.00 0.00
END
ENDFILE
```



**Figure 4.** KB2 program window

In the *Glass* option of *TriType*, the program introduces random node displacements in the initial logical mesh to create the disordered set of elements of Fig. 3. The *Smooth 0* command turns off the mesh smoothing that normally occurs after boundary fitting operations. This operation would wash out the initial perturbations. Disordered initial meshes are usually preferable for hydrodynamic simulations. Numerical errors can accumulate when mesh boundaries are lined up, resulting in a spurious shear plane (see Sect. 3).

The geometric sections describe six regions. Region 1 is a *Filled* region that represents the aluminum plate. Most of the specification lines define the irregular upstream boundary. After setting the boundaries, **Mesh** labels all nodes and enclosed elements as *RegNo* = 1 (blue triangles in Fig. 3). The remaining Open regions set the properties of nodes on the boundary of the solution volume. The top and bottom boundaries are *Sliding* surfaces that can move only in the direction of the unit vector (1.0,0.0). This condition is useful to represent symmetry boundaries. The nodes on the right-hand side of the solution-volume are *Pressure* points. They respond as though any adjacent elements outside the solution volume had an applied pressure.

Be sure you have saved the mesh by checking the status bar, and then exit **Mesh**. Next, run **KB2**. You should see the program window of Fig. 4. Click on the command *Edit input files* on the *File* menu and pick the file `RT_INSTAB.KIN`. This file, the control script for **KB2**, is listed below.

## KB2 input file RT\_INSTAB.KIN

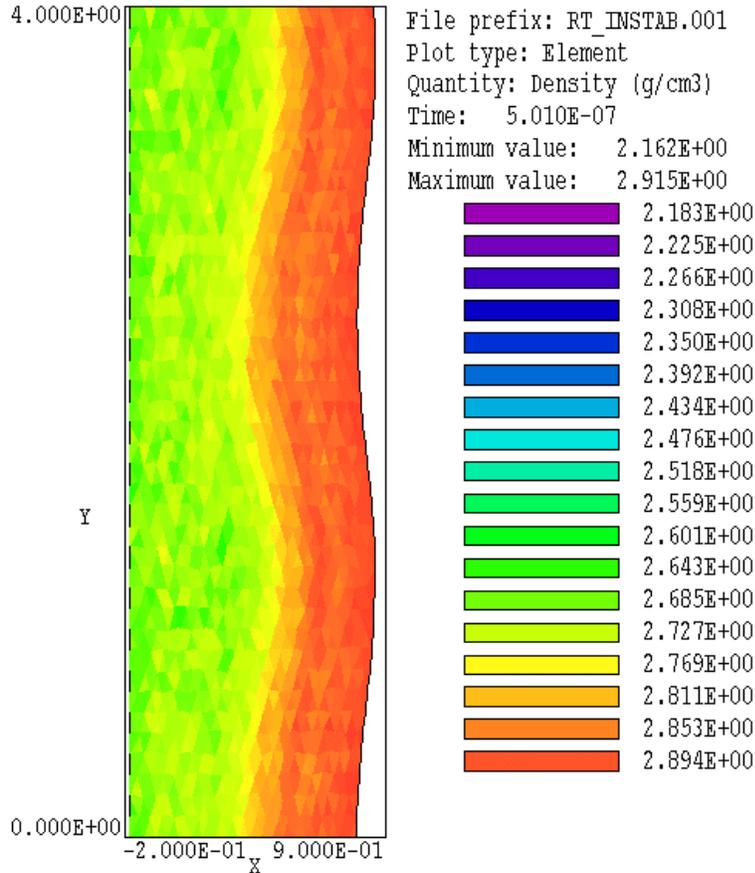
```
* RT_INSTAB.KIN
* ----- Run control -----
Geometry: Planar
Dt: 1.0E-9
TMax: 2.501E-6
DUnit: 1000.0
Interporder Second
Pressure 1 StepFunc.DRV 0.10E-6 5.0E9
* ----- Material properties -----
Material 1 KBT ALUM3715
* ----- Region properties -----
Region 1 Material 1
Viscosity 1 10.0
Region 2 Sliding 1.0 0.0
Region 2 Material 1
Region 3 Sliding 1.0 0.0
Region 3 Material 1
Region 4 Pressure 1
Region 4 Material 1
Region 5 Pressure 1
Region 5 Sliding 1.0 0.0
Region 5 Material 1
Region 6 Pressure 1
Region 6 Sliding 1.0 0.0
Region 6 Material 1
* ----- Diagnostics -----
DiagTime 0.0
DiagDt 0.5E-6
* Probes at minimum and maximum plate thickness
SetProbe 1.016 1.200
SetProbe 0.945 2.500
ProbeStep 20
* -----
EndFile
```



**Figure 5.** Plot created by **PROBE** for the RT\_INSTAB example

The commands of the control group set a maximum run time of 2.5 ?s and specify that the coordinates from **Mesh** should be converted from millimeters. The *Pressure* command loads and modifies a set of numbers from the data file `StepFunc.DRV` to define a step-function applied pressure that rises from 0.0 to 5.0 GPa in 0.1 ?s. The *Material* command loads the equation-of-state table `ALUM3715.KBT` as *MatNo* = 1. The *Region* commands set properties of the mesh regions and associate all of them with material *MatNo* = 1.

Exit the edit and click on the *Run* command. Pick the file `RT_INSTAB.KIN` in the dialog. The program initiates the run and reports progress in the screen window (Fig. 4). The run time will depend on the speed of your computer and should be a few minutes. After the run is completed, you can use Windows Explorer or your file manager to confirm that the program has created the following output files: `RT_INSTAB.KLS`, `RT_INSTAB.001`, ..., `RT_INSTAB006`, `RT_INSTAB_P01` and `RT_INSTAB.P02`. The



**Figure 6.** Density profiles at 0.5  $\mu$ s, RT\_INSTAB example

file RT\_INSTAB.KLS is a listing of run information that is often useful for debugging. The six files with numbered suffixes are data dumps at times 0.0, 0.5, ..., 2.5  $\mu$ s. The two files RT\_INSTAB.P01 and RT\_INSTAB.P02 are probe records at the two positions defined by the *SetProbe* commands.

The **PROBE** program is a utility to plot information recorded in probe files with the standard Field Precision format. It works with all initial-value programs in the **TriComp**, **KB**, **Maze** and **AMaze** series. The program is described in a separate manual. Run **PROBE** and click on the command *Load probe file*. Pick the file RT\_INSTAB.P01. Under Plot parameters choose the command *Set plotted quantity*. Pick the quantity vx or vz and you should see the plot of Fig. 5. It shows a general acceleration of the aluminum plate to about 2.3 km/s. The effect of shock reflections through the plate is apparent.

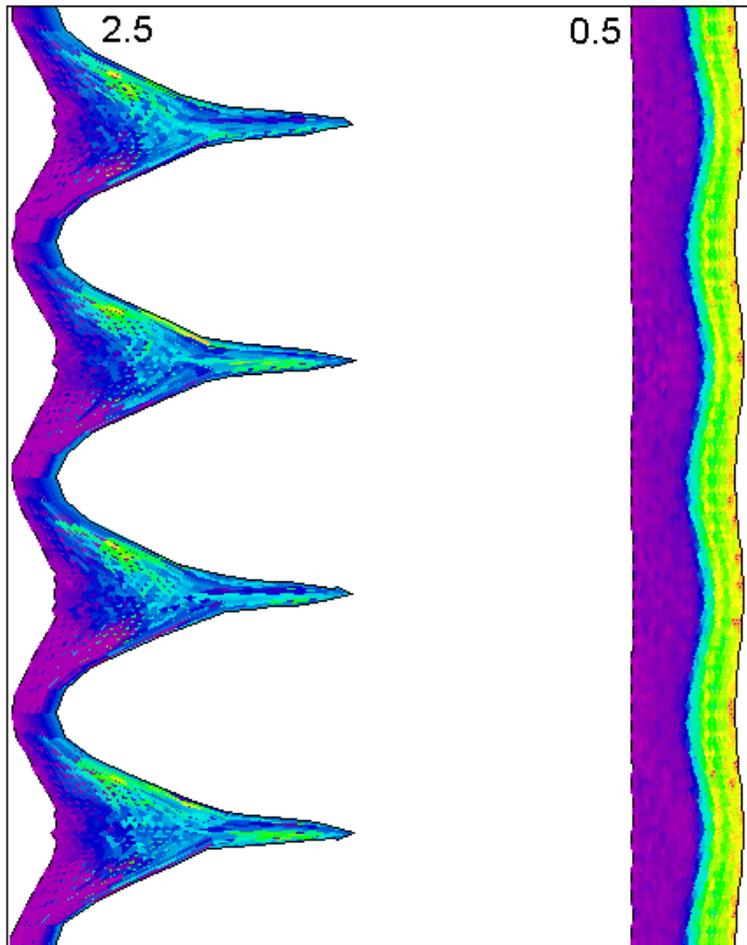


Figure 7. Outline of a 1 mm aluminum plate accelerated by a zero-mass fluid at 5 GPa. Abscissa:  $x_{\min} = -6.0$  mm,  $x_{\max} = 1.0$  mm. Ordinate:  $y_{\min} = 0.0$  mm,  $y_{\max} = 10.0$  mm.

The final step is to analyze the data dumps with the **KB2V** postprocessor (Fig. 1). Run the program and pick the command *Load first solution file* from the *File* menu. Choose `RT_INSTAB.001` and click *OK*. The program loads the first data file created at  $t = 0.0$   $\mu$ s before the application of the applied pressure. Click on *Plot type* in the *Spatial plot* menu and choose the *Region* option. Next, click on *Toggle element outline* in the *Spatial plot* menu. Using the *Zoom* and *Pan* functions you should be able to inspect the initial mesh and recreate the plot of Fig. 3.

Reset the plot type to *Element* and click on *Load next solution file* in the *File* menu. This loads the data dump created at  $0.5$   $\mu$ s. Click on *Plotted quantity* in the *Spatial plot* menu and choose *Density*. You

should see the plot of Fig. 6 which shows a compression wave created by the pulsed pressure moving through the plate. At this point, you can experiment with the plot and analysis capabilities of **KB2V**. Section 12 describes the program in detail. Figure 7 shows a plot constructed from the results of the `RT_INSTAB` example. The spatial limits of the pressure plots at 0.5 and 2.5  $\mu$ s have been adjusted to display the data in real space (Eulerian view). The plate accelerates to the left. The pressure boundary condition is equivalent to acceleration by a zero-mass fluid. In this case, the upstream boundary is subject to the Rayleigh-Taylor instability. The example illustrates the capability of **KB2** to follow the instability well into the non-linear regime.

The remainder of this section covers the operation of **KB2**. The second part is a detailed description of an of example of a hydrodynamic solution to introduce you to the features and organization of the software suite. The program `KB2 . EXE` can run in two modes:

# Interactive Windows program

# Autonomous program run from the Command Prompt, usually under batch file control

We shall first discuss the interactive mode. There are several ways to start `KB2 . EXE`: click the entry on the **KB** program launcher (Sect. 14), set up a shortcut on the Windows Desktop or Start menu, double-click on the program in Windows Explorer, or use the Run option on the Start menu. In all cases, the window shown in Fig. 2 appears. Entries in the pulldown menus have the following functions.

#### **Edit KIN file, Edit KLS file, Edit file**

The program has a built-in Windows editor that can be used to inspect input and output ASCII files. The difference between the commands is the choice of files displayed in the opening dialog. Files with suffix `KIN` are **KB2** input files. Files with suffix `KLS` are output listing files which contain run information and data dumps. Changing directories in the dialog does not change the working directory of the program. Close the editor to continue **KB2** operation.

#### **Start run**

This command initiates a **KB2** calculation. The dialog displays all files in the current working directory with the suffix `KIN`. Pick a file

and click *OK*. The run proceeds if the command script is syntactically correct and the required input data files are present. The data window (Figure 2) shows the run identifier, the number of cycles (time steps) processed, the simulation time, and the actual time elapsed.

### **Pause run**

**KB2** is a number cruncher that can occupy the full resources of your computer. As a result other programs may run slowly. If you need to perform a critical task in the middle of a **KB2** run, you can pause the program. Click the *Resume* button to continue with no loss of data.

### **Stop run**

The *Stop run* command is useful if a run is taking too long or you want to check output from the first part of an extended run. This command prompts for confirmation, closes all files and stops the program. Previously recorded data dumps and probe information will be available.

### **KB2 manual**

Displays a text version of this manual. The file `KB2 .HTM` must be in the same directory as `KB2 .EXE`.

You can run **KB2** as an autonomous program from the Command Prompt in Windows 2000 by supplying the prefix of the input file as a command line parameter. For example, suppose that the program is in the directory `\KB` and the input file `BLOCK03 .KIN` is in the directory `\KB\BUFFER`. Type

```
\KB\KB2 \KB\BUFFER\BLOCK03 <ENTER>
```

This capability is useful if you want to prepare a batch file for a set of extended runs. For example, the following file would perform five calculations with no operation intervention:

```
REM Cylindrical detonators
REM Variation of initiator radius
START /WAIT \KB\KB2 \KBDATA\CYLDET\CYLDET01
START /WAIT \KB\KB2 \KBDATA\CYLDET\CYLDET02
START /WAIT \KB\KB2 \KBDATA\CYLDET\CYLDET03
START /WAIT \KB\KB2 \KBDATA\CYLDET\CYLDET04
START /WAIT \KB\KB2 \KBDATA\CYLDET\CYLDET05
REM Job completed
```

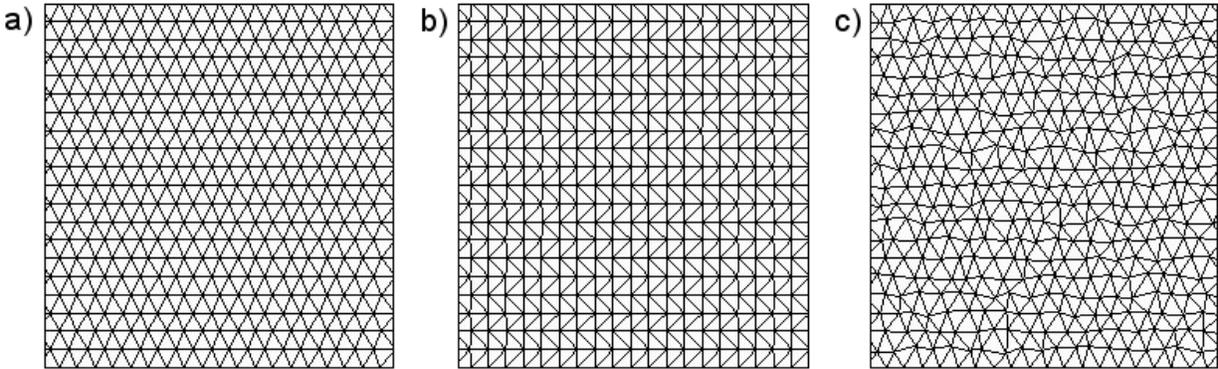
Note that Windows ME, which does not correctly support batch files in the command prompt mode. Windows ME will launch all tasks in the batch file simultaneously. This can lead to memory conflicts and incorrect sequencing of files. The following example illustrates a workaround to ensure that commands run sequentially:

```
ERASE CYLDET01.006
KB2 CYLDET01
:LOOP1
ECHO Checking that KB2 run is complete
IF NOT EXIST CYLDET01.006 GOTO LOOP1
ERASE CYLDET02.006
KB2 CYLDET02
:LOOP2
ECHO Checking that KB2 run is complete
IF NOT EXIST CYLDET02.006 GOTO LOOP2
```

In this case the file CYLDET02.006 is a data dump created near the end of the run. The check loops add a small time penalty. The *ECHO* commands should be included because their projection on the screen introduces a delay that reduces the number of checking loops. Do not put the command *ECHO OFF* at the beginning of the batch file or the run time will be significantly longer.

### Section 3. Mesh generation techniques

The **TriComp Mesh** program is used to define element divisions that represent the initial state of a physical system. In contrast to the programs of **TriComp**, the **KB2** mesh changes with time. Although programs such as **Pulse**, **TDiff** and **EMP** give time-dependent solutions for fields, the geometry of the mesh is fixed. The success of **KB2** solutions may depend on details of the initial mesh. **Mesh** script commands have been added to help in the creation of logical meshes for hydrodynamics. The first controls the general shape of elements.



**Figure 8.** Logical mesh types. a) *ISO*, b) *RIGHT*, c) *GLASS*.

### **TRITYPE [TriType] [RandDisp]**

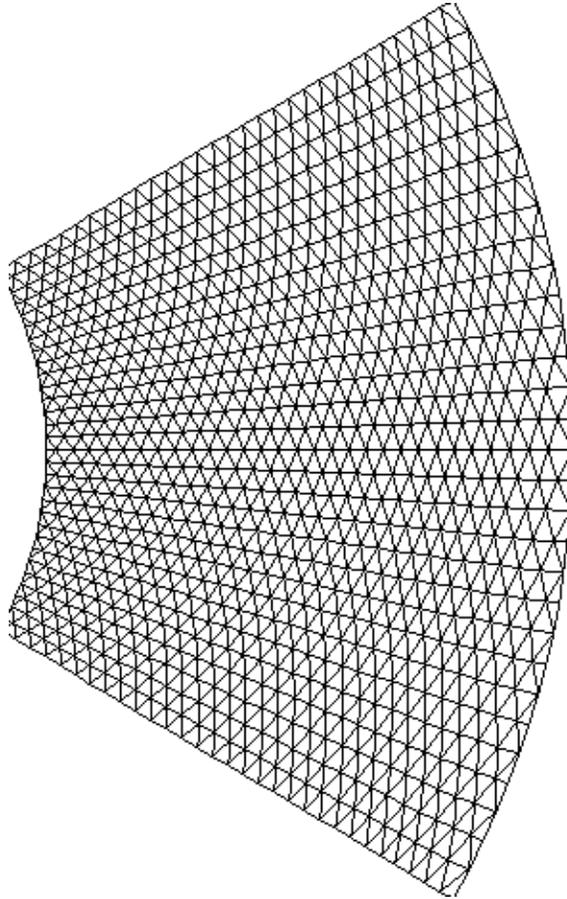
#### **TRITYPE GLASS 0.15**

**Comments:** This command sets the general shape of triangular elements. The string *ISO* denotes the standard row-oriented isosceles triangles that are generated by default (Fig. 8a). In response to the string *RIGHT*, Mesh fills the logical mesh region with right-angle triangles (Fig. 8b). The string *GLASS* gives a glasslike mesh. Here, random displacements are added at vertices to randomize the element sizes and shapes (Fig. 8c). This option can help remove interference effects resulting from coherent element motion parallel to the sides of a regular element array. An optional parameter *RandDisp* can be added to the *GLASS* option to control the size of the displacements. Values in the range 0.05 to 0.25 are useful. The default is  $RandDisp = 0.10$ .

**Parameters:** The string *TriType* can have the values *ISO*, *RIGHT* and *GLASS*. The optional real-number parameter *RandDisp* controls the randomization of node positions.

### **SMOOTH 0**

**Comments:** This command deactivates node smoothing after boundary fitting. It must be included when the *GLASS* triangle type is used. Otherwise, the final mesh will be smoothed to a set of isosceles triangles.



**Figure 9.** Logical mesh generated with the *Transform* command

**TRANSFORM TransType**  
**TRANSFORM CYLIN**

**Comments:** This command maps one logical mesh onto another.

**Parameters:** Presently, the only option for the string *TransType* is *CYLIN*.

A transformed mesh may be useful to represent the cross-section of an initially cylindrical object in planar geometry or a spherical shell in cylindrical geometry. The  $x$ -axis coordinates on the original logical mesh are interpreted as radii and the  $y$ -axis coordinates are interpreted as angles (in degrees). The transformed coordinates are:

$$x' = x \cos(y),$$

$$y' = x \sin(y).$$

Figure 9 shows the logical mesh generated by the following commands:

```

Transform Cylin
XMesh
    2.00  5.00  0.10
End
YMesh
    -30.0  30.0  2.0
End

```

## Section 4. Structure of the KB2 command script

**KB2** may use and create several types of files. Table 4.1 summarizes the function of different files. All files are in ASCII format. In this section and the following ones, we shall concentrate on the preparation of input files to control the run and to supply data. The input script is the fundamental control file required for all **KB2** runs. It is an ASCII text file that consists of a list of commands with parameters, similar to a BASIC interpreter program. The main difference is that the commands can appear in any order - the program sorts out the logic before proceeding to execution. The command file must have a name of the form `FPrefix.KIN`, where `FPrefix` is any valid filename from 1 to 20 characters in length. The script and other input files for mesh geometry, equation-of-state data and pressure waveforms should be in the same directory.

Section 2 discussed a sample input script. The file may contain command, comment and blank lines. A command line consists of a keyword and one or more parameters. Commands may appear in any order because **KB2** sorts and processes information after all commands are read. You can enter commands and parameters in upper or lower case. Comment lines must begin with '\*' (asterisk-space). Commands and parameters can be indented. The program ignores spaces and other delimiters. The following are valid delimiters.

Space: ' '  
 Comma: ','  
 Tab character  
 Colon: ':'  
 Equal sign: '='  
 Left parenthesis: '('  
 Right parenthesis: ')'

<b>Table 4.1. KB2 input and output files</b>	
<b>Name form</b>	<b>Function</b>
FPrefix.MIN	Input file for the Mesh program, description of the initial system geometry
FPrefix.MLS	Listing of information and error messages from Mesh
FPrefix.MOU	Mesh specification file created by Mesh, required input for KB2
FPrefix.KIN	Required input script controls operation of KB2
PressName	Input table of waveform values for an applied pressure
KBTName.KBT	Input equation-of-state table for a KBT material
FPrefix.KLS	KB2 output listing file with extensive run information and data
FPrefix.001	KB2 output data dumps for <b>KB2V</b> analyses
FPrefix.P01	Files of temporal information for a given element for Probe analyses
GPrefix.DAT	Data file created during a <b>KB2V</b> analysis session

Command input terminates with the *EndFile* statement. The command signals that input information is complete and **KB2** should begin execution. You can enter descriptive text in any format after *EndFile*.

Commands fall into seven categories:

#### **Controls**

Set general properties like the maximum run time.

#### **Materials**

Define material models to be associated with regions and elements

**Regions**

Set region properties, such as the initial velocity or hydrodynamic state

**Diagnostics**

Control how much information is written to the listing, data dump and probe files

**Section 5.  
Program  
control  
commands**

Control commands can appear anywhere in the script file, although it is good practice to group them at the beginning. This section shows each command in symbolic form and also as it might appear in the script.

**GEOMETRY [PLANAR, CYLINDRICAL]  
GEOMETRY CYLINDRICAL**

**Comments:** This required command sets the solution symmetry.

**KB2** can perform calculations in planar geometry or cylindrical. In the first case, quantities may vary in the  $x$ - $y$  plane and are uniform in  $z$ . In cylindrical geometry, quantities vary in  $r$ - $z$  but are azimuthally symmetric.

**Parameters:** A single parameter, either the string *Planar* or *Cylindrical*.

**DUNIT DUnit  
DUNIT 100.0**

**Comments.** Set units for converting dimensions in the Mesh input file. The quantity DUnit equals the number of Mesh units per meter. For example, if dimensions in the mesh file are in mm, set DUnit = 1000.0

**Parameters:** The real-number value of DUnit.

**TITLE Run title string  
TITLE Ignition of TNT rod (small ignitor)**

**Comments:** The title string is useful for archiving a run and appears in plots created by **KB2V**.

**Parameters:** A string of up to 60 characters

**TMAX TMax****TMAX 5.75E-6**

**Comments:** Sets the maximum run time.

**Parameters:** A single real-number quantity, the maximum run time in seconds.

**INTERPORDER [First, Second]****INTERPORDER First**

**Comments:** This command controls the type of interpolations performed for KBT materials. First order interpolations are generally sufficient for shock solutions at high temperature or pressure. Second order interpolations increase the run time significantly, but may be necessary to achieve accuracy in simulations where there are small changes of density or pressure. This command has no effect on *ShockEOS*, *Gamma* and *Explosive* materials.

**Parameters:** A single string, either *First* or *Second*. The default is *First*.

**ERRSTATUS [All,Partial,None]****ERRSTATUS Partial**

**Comments:** **KB2** may perform millions of interpolations in runs with KBT materials. The operations are of the form  $T(?U)$  and  $P(?T)$ . This command controls how the code responds when it detects an interpolation error. Errors in  $T(?U)$  may occur if the material is near ambient temperature and if the KBT table has widely-spaced values at low temperature. In this case, there is little error introduced if we simply set the temperature equal to the initial value for the region if an error is detected. In the *Partial* option, the program ignores errors of the type  $T(?U)$ , but terminates following an error in  $P(?T)$ . In the *All* option, the program stops with an error message for any interpolation error. In the *None* option, the program ignores all interpolation errors. This option is useful mainly for debugging. This command has no effect for *ShockEOS*, *Gamma* and *Explosive* materials.

**Parameters:** A single string, either *All*, *Partial* or *None*. The default is *Partial*.

**SAFETY Safety****SAFETY 5.0**

**Comments:** **KB2** can pick a time step automatically for simple runs where there are no abrupt changes in quantities (i.e., collisions

between regions). This command sets a safety factor for estimating a stable time step from the current element properties. Higher numbers are more conservative. Lower numbers reduce the run time at the risk of a numerical instability.

**Parameters:** A single real-number quantity for the *Safety* variable. The default value is 10.0.

## **NCHECK NCheck**

### **NCHECK 50**

**Comments:** Sets the number of program steps between recomputations under automatic time step control.

**Parameter:** Single integer parameter, *NCheck*. The default value is 5.

## **DT Dt**

### **DT 1.0E-09**

**Comments:** With this command you can set the value of the initial time step and disable automatic time step selection. The value must be short enough to preserve hydrodynamic stability. A general criterion is

that the time step should be shorter than the minimum element length (along a side) divided by the maximum velocity. The velocity may correspond to an initial directed velocity or to the propagation velocity of shocks. You may need to experiment to find a value of *Dt* that gives numerical stability with an acceptable run time.

**Parameter:** A single real-number parameter, *Dt* in seconds.

## **DTCHANGE TChange DtNew**

### **DTCHANGE 5.0E-06 2.5E-09**

**Comments:** The time step *Dt* changes to the value *DtNew* at time *TChange*. This command allows you to fine-tune the program for difficult simulations. For example, a short time step may be necessary during material collision times of the convergence of shock on a symmetry axis. The *DTChange* command over-rides values set by the *Dt* command or automatic step selection.

**Parameters:** Two real-number parameters, both in seconds. The first (*TChange*) is the approximate time at which the change should occur, and the second (*DtNew*) is the new value of the time step.

## Section 6. Applied pressure waveforms

There are three ways to drive shock hydrodynamic solutions in **KB2**: setting initial velocities between colliding regions, detonating explosives, and applying a time-dependent drive pressure. This section explains how to define a time-dependent pressure waveform that acts on the nodes of one or more line regions at the boundary of the solution volume. The controlling command has the following form:

### **PRESSURE TabNo TabName**

#### **PRESSURE 3 LinDrive.PRS**

**Comments:** Up to five *Pressure* commands can appear in the input script file to define applied pressure waveforms that may be applied on different boundary regions.

**Parameters:** The quantity *TabNo* is a reference number for the table, an integer in the range 1-5. The string parameter gives the full name of a tabular function file in the current data directory. The file format is described below.

Arbitrary time variations of pressure can specified with tabular functions that follow the same format in all Field Precision programs. Tabular function are ASCII files that contain up to 256 data lines. Each data line contains two real numbers separated by spaces or other valid delimiters (see Sect. 3). The first number in each line is the time (in s) and the second number is a corresponding value of the pressure (in Pa). The time interval need not be uniform. For example, you may want to cluster values near a time when there is a sharp transition of pressure. The *EndFile* command marks the end of data entry. Tables can also contain comment lines starting with '\*' (asterisk-space) and blank lines. The following illustrates a normalized pressure table:

```
* Normalized smooth step function
* Rise to 1.0 at time 1.0
* =====
      0.00000      0.00000
      0.05000      0.00616
      0.10000      0.02447
      0.15000      0.05450
      0.20000      0.09549
      0.25000      0.14645
      0.30000      0.20611
      0.35000      0.27300
      0.40000      0.34549
      0.45000      0.42178
      0.50000      0.50000
      0.55000      0.57822
```

0.60000	0.65451
0.65000	0.72700
0.70000	0.79389
0.75000	0.85355
0.80000	0.90451
0.85000	0.94550
0.90000	0.97553
0.95000	0.99384
1.00000	1.00000
1.05000	1.00000
1.10000	1.00000
1.50000	1.00000
2.00000	1.00000
5.00000	1.00000
10.00000	1.00000
20.00000	1.00000
50.00000	1.00000
100.00000	1.00000

ENDFILE

**KB2** finds interpolated values using a cubic spline interpolation of the data. This method is very flexible, but may give unexpected results for noisy or inaccurate data. For this reason, the program writes a test series of 50 interpolations uniformly spaced over the time interval 0.0 to *TMax* in the listing file.

An alternate form of the *Pressure* command allows you to add multiplying factors to adjust the values of normalized tables.

**PRESSURE TabNo TabName TMult PMult**  
**PRESSURE 3 LinDrive.PRS 1.0E-06 25.0E09**

**Comments:** With this form of the Pressure command time values are multiplied by TMult and pressure values by PMult when they are loaded into the program. This feature makes it convenient to maintain a library of normalized waveforms, using multiplication factors to apply them to different simulations. Note that the test interpolation listing shows values as they will be used in the program.

**Parameters:** The quantity *TabNo* is a reference number for the table, an integer in the range 1-5. The string parameter gives the full name of a tabular function file in the current data directory. The first real-number parameter following the table name is a multiplication factor for time values. The values used in the program are the table values multiplied by *TMult*. The second real-number parameter is a multiplication factor for the pressure values.

## Section 7. Defining standard materials

The manual *Physics of the KB Codes* discusses the material models used in **KB1** and **KB2**. It is important to note that the programs are useful for shock phenomena at very high temperatures and pressures. The current versions do include support for elastic or inelastic material strength and do not include effects of energy transport by thermal conduction or radiation. **KB2** uses four models for the hydrodynamic properties of materials. They differ in how element pressure is inferred as a function of density and internal energy.

### **KBT Materials**

For these materials the programs use the KBT tables to find both pressure and temperature of the material. The tables are based on the Sesame tables distributed by Los Alamos National Laboratory (see the manual *KB Tables* for a complete description). The *KBT* model has several advantages compared to the *Gamma* and *ShockEOS* models:

# It includes effects of phase changes in materials

# It gives information on temperature as well as density, pressure and specific energy

# It correctly represents both shock compression and subsequent expansion of materials

The disadvantage is that the interpolations require considerably more run time than the simple formula evaluation of the *ShockEOS* and *Gamma* models.

### **Shock EOS materials**

The shock equation-of-state model gives useful approximations for materials that are not covered by the KBT tables. The model yields changes in pressure as a function of density and internal energy with the assumption that material states lie on a shock Hugoniot curve. The manual *Shock EOS Tables* gives detailed information. The model is applicable to materials undergoing shock compression, but it does not provide an accurate representation of subsequent expansion.

### **Gamma law materials**

Many gases follow the law  $p = (\gamma - 1) \rho_o U$  over a wide range of density and pressure. If applicable, the model has two advantages

over interpolations on the KB tables.

# There are fewer numerical operations, leading to shorter run times.

# There is less chance of an interpolation error under extreme conditions.

The gamma law model is useful to avoid numerical instabilities in treating materials with very high density compressions, such as a gas entrapped between colliding solid regions.

### **Explosive materials**

Explosive materials use both the *ShockEOS* and *Gamma* models. The shock equation-of-state applies to the solid or liquid materials before detonation. An element of explosive material detonates if it is subjected to a shock that raises the pressure above a critical value. At detonation, the density is unchanged while the internal energy increases by  $Q$  (the specific energy of detonation). For subsequent times the increased pressure is calculated from the gamma law model.

The following commands set the properties of a numbered material model. You can define up to 20 material models. A model can be associated with several regions with different initial conditions.

### **MATERIAL MatNo KBT TabName**

#### **MATERIAL 5 KBT ALUM3715**

**Comments:** Loads the KBT table *TabName*.KBT and assigns it as material model *MatNo*. The table must be in the current directory. The initial hydrodynamic state of the material can be set individually in different regions.

**Parameters:** The parameter *MatNo* is an integer in the range 1-20 to identify the material model. The string *KBT* designates that the equation-of-state will be determined from interpolations on a KB table. The string *TabName* (1-8 characters) is the prefix of the KB table.

### **MATERIAL MatNo SHOCK C0 S1 S2**

#### **MATERIAL 4 SHOCK 2.985E3 0.5148 2.814E-4**

**Comments:** Defines material *MatNo* as a *ShockEOS* material and sets the relationship between  $u_s$  (shock velocity) and  $u_p$  (particle velocity) as defined in the manual *ShockEOS Tables*,

$u_s = C_0 + S_1 u_p + S_2 u_p^2$ . The initial density can be set individually in different regions. The initial pressure equals zero.  
**Parameters:** The parameter *MatNo* is an integer in the range 1-20 to identify the material model. The string *Shock* designates that the equation-of-state will be determined from the Shock EOS model. The three real-number parameters are  $C_0$  (m/s),  $S_1$  and  $S_2$  (s/m).

### **MATERIAL MatNo GAMMA Gamma**

#### **MATERIAL 3 GAMMA 2.53**

**Comments:** Defines material *MatNo* as an gamma-law material where pressure  $p$  is related to internal energy  $U$  by  $p = (\gamma - 1)\rho_o U$ . The initial density and internal energy of the material in a particular region can be set with *Region* commands.  
**Parameters:** The parameter *MatNo* is an integer in the range 1-20 to identify the material model. The string *Gamma* designates that the equation-of-state will be determined from the gamma-law model. The real number parameter is the  $\gamma$  value.

## **Section 8. Defining explosive materials**

This section covers the form of the *Material* command that defines the **KB** parametric model for explosives (discussed in Sect. 4 of the manual *Physics of the KB Codes*.) A typical simulation starts with the creation of a shock wave in the explosive by an initiator. The shock could be produced by collisions between regions, a rapid local heating, or the influence of an adjacent explosive. During the ignition phase it is sufficient to treat the solid explosive with the shock EOS model (see Sect. 3 of *Physics of the KB Codes*). If the shock pressure in an element exceeds a minimum value, *PInit*, the element material undergoes a rapid chemical change. The simplifying assumption in **KB** is that the internal energy of the element instantaneously increases by  $Q$  (the specific heat of explosion) and that the material changes to a gaseous state. The detonation products are described by an ideal  $\gamma$ -law equation-of-state. Although the model is relatively simple, it has two advantages: 1) the required material data are generally available and 2) the results are in good agreement with theoretical and experimental values for detonation front velocities.

Note that results are not sensitive to the exact choice of the shock EOS parameters  $C_0$ ,  $S_1$  and  $S_2$ . If you do not have values for a specific material or initial density, you can usually get satisfactory results with

parameters of a material with similar properties. The quantity  $P_{Init}$  is not an absolute property of an explosive, but rather a parameter that can be adjusted to model different types of behavior. Values just below the Chapman-Jouguet pressure give well-defined detonation fronts. Lower values correspond to more sensitive explosives, such as inhomogeneous materials with potential hot spots.

**MATERIAL MatNo EXPL C0 S1 S2 PInit Q Gamma [INIT]  
MATERIAL 5 EXPL 2461.0 3.517 -8.810E-04 2.0E10  
5.183E+06 2.63**

**Comments:** The initial density and internal energy of the material in a particular region can be set with *Region* commands of Sect. 10.

**Parameters:** The parameter *MatNo* is an integer in the range 1-20 to identify the material model. The string *Expl* designates that the equation-of-state will be determined from the explosives model. There are six real-number parameters. The quantities  $C_0$  (m/s),  $S_1$  and  $S_2$  (s/m) define the shock equation-of-state for the undetonated material. The quantity  $P_{Init}$  (Pa) is a threshold pressure for detonation,  $Q$  (J/kg) is the specific energy released in the explosion, and  $\gamma$  defines the equation-of-state for the gaseous detonation products. If the string parameter *Init* is included, the material will detonate at a  $t = 0.0$ . This option is useful for checking the safety of explosive configurations. Self-ignited material should not cause a detonation wave.

## Section 9. Regions - element properties

The commands in this section set the properties of elements in regions with non-zero volume. Some of the commands may also apply to the nodes of both filled and open regions. Node commands are described in the following section.

**REGION 5 MATERIAL 3  
REGION RegNo MATERIAL MatNo**

This required command associates elements and nodes marked by region number *RegNo* with the properties of material number *MatNo*.

### **HYDINIT 3 4500.00 530.0**

#### **HYDINIT RegNo Rho Temperature**

Although two regions may be associated with the same material model, they can be assigned different initial hydrodynamic conditions. This form of the command sets the initial density ( $\text{kg/m}^3$ ) and temperature ( $^{\circ}\text{K}$ ) for regions associated with KBT materials. The program attempts to set the pressure and internal energy by interpolations on the tables. You can check the values in the KLS file. If the program has trouble interpolating on the low end of the table, you can usually find acceptable parameters using the **KBView** utility. The initial conditions of all regions with non-zero volume must be defined.

### **HYDINIT 4 4500.0 0.0**

#### **HYDINIT RegNo Rho Epsi**

This form of the command sets the initial density ( $\text{kg/m}^3$ ) and internal energy ( $\text{J/kg}$ ) for regions associated with *Gamma*, *Shock* and *Expl* materials. The initial conditions of all regions with non-zero volume must be defined.

### **VISCOSITY 5 0.25**

#### **VISCOSITY RegNo CVis**

This commands define a value of the artificial viscosity coefficient for elements with region number *RegNo*. The default value is 10.0. Assign higher values if there are spurious oscillations at shock fronts and low values if shock fronts spread over several elements.

## **Section 10. Regions - node properties**

The commands in this section set the property of nodes. The *VelInit* command applies to both *Filled* (non-zero volume) or *Open* (line) regions. The other commands apply only to *Open* regions and are used to set boundary conditions.

### **VELINIT 3 Cart 0.000 -3.0E3**

#### **VELINIT RegNo CART vx vy**

This command sets an initial velocity for all nodes associated with region *RegNo*. The string *CART* indicates that all nodes have the same velocity defined by  $[\text{vx}, \text{vy}]$  or  $[\text{vz}, \text{vr}]$ . Enter the velocity values in units of m/s.

### **VELINIT 3 RADIAL 0.000 0.000 -3.0E3**

#### **VELINIT RegNo RADIAL XConv YConv MagV**

This command sets an initial velocity for all nodes associated with region *RegNo*. The string *RADIAL* indicates that nodes have velocities assigned to define a medium converging toward a point. The compression is cylindrical in *Planar* solutions, In *Cylindrical* solutions, the compression is spherical if the reference point is on the *z*-axis. The quantities *XConv* and *YConv* define the convergence point (x,y) or (z,r). Enter the values in units set by *DUnit*. The quantity *MagV* is the magnitude of the velocity assigned to all nodes in units of m/s..

### **REGION 3 FIXED**

#### **REGION RegNo FIXED**

This command assigns the *Fixed* (immovable) condition to nodes associated with region number *RegNo*. The condition is useful for symmetry boundaries and usually applies to line regions. A surface is immovable if 1) there are equal normal forces on both sides and 2) the tangential force is zero. The *Fixed* condition may also apply if a surface is adjacent to a material with infinite density.

### **REGION 7 SLIDING 0.7071 0.7071**

#### **REGION RegNo SLIDING ux uy**

The *Sliding* boundary condition is useful for modeling symmetry boundaries in a periodic system. Along such a boundary the normal force is zero but the tangential force may be non-zero. In response to such forces, the nodes can move only along the line defined by the unit vectors [ux,uy] (or [uz,ur] in cylindrical problems). The condition applies only to line regions. The boundary should be a straight line aligned with the normal vector.

### **REGION 4 PRESSURE 2**

#### **REGION RegNo PRESSURE TabNo**

This command sets nodes associated with region number *RegNo* as *Pressure* nodes and associates them with pressure table *TabNo* (see the *Pressure* command). Pressure regions must be line regions on the periphery of the solution volume. If a node has the *Pressure* attribute, **KB2** assigns the pressure from the associated table calculated at the current time to all adjacent elements with *NReg* = 0. In other words, the pressure is applied to elements (real or fictitious) outside the solution volume. In this way, time-varying pressures can be applied along boundary regions of arbitrary shape.

The example `RT_INSTAB` discussed in Sect. 2 is a good example of how to implement *Pressure* and *Sliding* boundaries. The following script file lines control the features:

```
01: Pressure 1 StepFunc.DRV 0.10E-6 5.0E9
02: Material 1 KBT ALUM3715

03: Region 1 Material 1
04: Viscosity 1 10.0

05: Region 2 Sliding 1.0 0.0
06: Region 2 Material 1

07: Region 3 Sliding 1.0 0.0
08: Region 3 Material 1

09: Region 4 Pressure 1
10: Region 4 Material 1

11: Region 5 Pressure 1
12: Region 5 Sliding 1.0 0.0
13: Region 5 Material 1

14: Region 6 Pressure 1
15: Region 6 Sliding 1.0 0.0
16: Region 6 Material 1
```

Line 1 loads data on the time variation of pressure from the normalized file `StepFunc.DRV`. The time values are multiplied by  $0.10 \times 10^{-6}$  to give a step-function risetime of 0.1  $\mu$ s. The pressure values are multiplied by  $5.0 \times 10^9$  to give a peak pressure of 5 GPa. The data are referenced as *Pressure Table 1*. Line 2 sets *Material Number 1* as a *KBT* type and loads equation-of-state data. Line 3 associates *Region 1* with *Material 1*, while Line 4 sets a value of artificial viscosity in the elements (see Sect. 9). Regions 2 and 3 are line regions at the top and bottom of the solution representing symmetry boundaries of the periodic system. They are free to move in the *x*-direction but cannot move in *y*. Lines 5 and 7 specify this condition. Note that there are also *Material* commands (Lines 6 and 8) for the region. They specify that the boundary nodes should be advanced like internal nodes surrounded by elements of *Material 1* with corrections for the sliding condition. In the absence of these commands, the program would not advance the positions and velocities of the boundary nodes.

Region 4 (Line 9) is the boundary on the right-hand side of the plate on which the pressure is applied. Note that a *Material* assignment command (Line 10) is included so that the node will be advanced in velocity and position. Regions 5 and 6 are single nodes at the top and bottom of the right-hand boundary. The special regions must be defined because the nodes, in contrast to those of Regions 2,3 or 4, must have both the *Sliding* and *Pressure* conditions.

The example BLOCK03 illustrates how to set up *Sliding* boundaries for a solution with multiple materials with different initial velocities.

## Section 11. Diagnostic commands

The commands of this section control the data files created by **KB2**. There are two types of information available from an initial-value simulations: 1) records of quantities over the full solution volume at specified times (spatial data dumps) and 2) records of quantities at all time steps in specified elements (probes). Three commands control the creation of spatial dumps.

### **DIAGDT DTime** **DIAGDT 5.0E-6**

**Comments:** Write a data dump at approximately uniform time intervals. Because the time step may vary, the record may not occur at an exactly uniform interval. The convention is that **KB2** writes the spatial information at or immediately after the specified time.

**Parameters:** The real-number parameter *DTime* is the approximate time interval in seconds.

### **DIAGTIME TDiag** **DIAGTIME 7.333E-8**

**Comments:** You can set up to 50 specific times to make data dumps. For example, you may want to create a detailed set of views at the collision time of two regions. This command can work in conjunction with the *DiagDt* command. The data dump occurs at or immediately after the specified time. The *DiagTime* commands must appear in the input script in chronological order (*i.e.*, ordered from the earliest to the latest data dump).

**Parameters:** Enter the time *TDiag* in seconds.

## DIAGSTEP NDiag DIAGSTEP 200

**Comments:** When this command is issued, **KB2** writes data dumps at intervals separated by a given number of time steps. This command works in conjunction with the *DiagDt* and *DiagTime* commands. Be careful entering dump information, or you may fill a hard disk with data.

**Parameters:** The integer parameter *NDiag* is the number of time steps between data dumps.

The final two commands control the placement of probes in the solution volume. **KB2** can create from 1 to 12 probe files with names of the form *RunName.P01*, *RunName.P02*, .... Information in the files can be plotted or analyzed with the **Probe** program.

## SETPROBE XPos YPos SETPROBE 0.50 5.67

**Comments:** Probes are located at the center of an element. The output files list the corresponding element quantities (density, pressure, ...). Node quantities (position, velocity) are evaluated at the element center-of-mass.

**Parameters:** Two real numbers that give the approximate initial position of the probe. Enter (x,y) or (r,z) in units set by *DUnit*. The program locates the element centroid closest to the given point.

## PROBESTEP NStepProbe PROBESTEP 5

**Comments:** Simulations with small values of *Dt* may run for thousands of time steps. In this case, there is no need to record probe information at each step because only 200-300 points are sufficient for a good plot. This command instructs **KB2** to make entries in probe files at intervals. For example, if *NStepProbe* = 5 the code makes a record every fifth time step. The purpose of the *ProbeStep* command is to keep probe files at a manageable length. You can make long files if you want. The **Probe** program has an automatic interval feature so that the number of loaded points does not exceed the dimensions of the plot array.

**Parameters:** The integer parameter *NStepProbe* is the number of code time steps per probe file record. The default is *NStepProbe* = 1.

## Section 12. Features of KB2V

**KB2V** (for View KB2) is the interactive graphical analysis program for output files from **KB2**. General features of the program are described in the *TriComp Solution Programs and Postprocessors Manual*. This section summarizes special features for hydrodynamic analyses.

### Program units

Spatial dimensions: meters or units set by *DUnit*.  
Density: ? (gm/cm<sup>3</sup>)  
Pressure: P (GPa)  
Temperature: °K  
Internal energy: MJ/kg  
Velocities: km/s

### Contour plot quantities

Density: ? (gm/cm<sup>3</sup>)  
Pressure: P (GPa)  
Temperature: °K  
Internal energy: MJ/kg  
|V|: km/s  
V<sub>x</sub>, V<sub>z</sub>: km/s  
V<sub>y</sub>, V<sub>r</sub>: km/s

### Element plot quantities

Density: ? (gm/cm<sup>3</sup>)  
Pressure: P (GPa)  
Temperature: °K  
Internal energy: MJ/kg  
|V|: km/s  
V<sub>x</sub>, V<sub>z</sub>: km/s  
V<sub>y</sub>, V<sub>r</sub>: km/s

### Vector plot quantities

V<sub>x</sub>, V<sub>z</sub>: km/s  
V<sub>y</sub>, V<sub>r</sub>: km/s

### Surface plot quantities

Density:  $\rho$  (gm/cm<sup>3</sup>)  
Pressure: P (GPa)  
Temperature: °K  
Internal energy: MJ/kg  
|V|: km/s  
V<sub>x</sub>, V<sub>z</sub>: km/s  
V<sub>y</sub>, V<sub>r</sub>: km/s

### Line scan quantities

Density:  $\rho$  (gm/cm<sup>3</sup>)  
Pressure: P (GPa)  
Temperature: °K  
Internal energy: MJ/kg  
|V|: km/s  
V<sub>x</sub>, V<sub>z</sub>: km/s  
V<sub>y</sub>, V<sub>r</sub>: km/s

### Volume integrals

The following analyses are performed in response to the *Volume integrals* command. Data are listed by region and for the full solution volume.

- 1) Cross-sectional area (planar) in m<sup>2</sup> or volume (cylindrical) in m<sup>3</sup>.
- 2) Kinetic energy, the volume integral of  $\frac{1}{2}\rho V^2$  in J/m (planar) or J (cylindrical).
- 3) Internal energy, the volume integral of  $\rho e$  in J/m (planar) or J (cylindrical).

### Element properties

In response to the *Element properties* command, **KB2V** lists the following properties: region and materials number, average position and indices.

### Region properties

In response to the *Region properties* command, **KB2V** performs integrals over all elements of the region and lists the following quantities: material number, number of elements, area (planar) or volume (cylindrical), kinetic energy, internal energy, average position, average velocity, mass, average density, average pressure and expansion rate. For planar solutions, the expansion rate is the time rate of change for the area. It equals the time rate of change of the volume for cylindrical solutions.

### Line integrals

The program calculates two integrals over the given line. The first is the integral of the normal pressure force. The units are N/m in planar solutions and N in cylindrical solutions. The second quantity is the mass flow through the line in units of kg/m-s (planar) or kg/s (cylindrical).

## Section 13. KB program launcher

The **KB** software system incorporates several programs. The **KB** Program Launcher is a convenient utility to set a standard data directory and to run any of the **KB** programs as sub-tasks.

You can set up a shortcut to **KB . EXE** on the Windows Desktop or in the Start menu. When you run the program, a floating dialog box (Fig. 2) is displayed. The *Program directory* is the location where **KB** looks for executable programs. If you have installed the programs (including **KB . EXE**) in the directory **\KB**, the program directory is automatically set. If the program directory is incorrect, choose *Set program directory* and then use the dialog box to transfer to the directory that contains the **KB** programs. **KB** will record the new location and set the correct program directory the next time it runs.

The next step is to define a data directory. Use the *Set data directory* command to move to the location where input and output files should be stored. In the default configuration, the directory is **\KB\BUFFER**. **KB** records this location and automatically sets the data directory the next time it runs.

From this point, program operation is simple. Click on the activate button next to the program you want to run. After starting, the selected program will be ready to seek input files from the assigned data directory. You can change the data directory at any time. If a program is missing, nothing will happen when you press the corresponding button.