

## THE TOOL FOR HIGH-VELOCITY INTERACTION AND DAMAGE OF SOLIDS

E. I. KRAUS AND I. I. SHABALIN

Khristianovich Institute of Theoretical and Applied Mechanics of the Siberian Branch of the Russian  
Academy of Sciences  
Institutskaya 4/1, 630090 Novosibirsk, Russia  
e-mail: kraus@itam.nsc.ru

**Summary.** The main points of development of numerical tools for the simulation of deformation and failure of complex technical objects under non-stationary conditions of extreme loading are presented. The possibility of extending the dynamic method for construction of difference grids to the 3D case is shown. A 3D realization of discrete-continuum approach to the deformation and failure of complex technical objects is carried out. The efficiency of the existing software package for 3D modeling is shown.

### 1 INTRODUCTION

Numerical modeling of the behavior of matter under the influence of shock load requires taking into account the laws of elastic-plastic flow and, therefore, the stress and strain. Such problems are solved using the equations of conservation of mass, momentum, and energy [1–7]. The physics of the processes at high-speed collision of solid bodies can be studied in one-dimensional case, for example, the construction of the equations of state of materials. However, there is a need for both two-dimensional and spatial calculations for modeling the processes of deformation and destruction of complex technical objects that do not fit into one-dimensional schemes. Thus, the phenomenon of rebound of nutating cylinder off an obstacle in a collision at an angle or impact of a complex technical object, for example, model reactor at its center line due to its multiply connected structure are impossible to describe by the two-dimensional numerical tools. Therefore, there is an urgent need for further development of the existing two-dimensional numerical tools [8] to the spatial case.

The software package is designed to solve the time-dependent problems of deformation and fracture of materials at loading of complex technical objects. The package implements continuous-discrete approach, combining Lagrangian coordinates for the solid material and discrete particles of finite size simulating fragments of the fractured material. The behavior of the materials is described by elastoplastic model with a few-parameter equation of state. The difference ratios are put into a tetrahedral grid, which is constructed in a dynamic way in complex, multiply connected computational domains. All this allows one to simulate the processes of impact loading of complex technical objects to “the end”, i.e. until the complete absorption of the kinetic energy of the projectile body and (or) splitting of the object into separate fragments.

**2010 Mathematics Subject Classification:** 74D10, 74R15, 65K05.

**Key words and phrases:** equation of state, computer simulation, high velocity impact.

The software package developed for the spatial modeling includes:

- equations of mass balance, momentum, and energy [9, 10];
- setting the initial data and boundary conditions for all elements of the object [3, 9, 10];
- equations of state and equations of the process for the reactor materials [11–15];
- necessary criteria for the destruction of materials, covering a variety of mechanisms of their destruction [3, 6, 8];
- algorithm for replacing cells containing the fractured material by discrete particles of finite size [3, 8, 10];
- extending the methods for dynamic construction of the difference grid [8, 16] to the spatial case;
- finite-difference equations for the balance equations on a tetrahedral grid [10];
- symmetric algorithm for calculating contact boundaries and condition for stable solving of the system of difference equations [8, 10].

Let us take a more detailed look at the above steps.

## 2 FORMULATION OF THE PROBLEM

We start from the Lagrangian method of describing the environment as the most appropriate way to describe the interaction of deformable solids [8]. The model consists of equations of motion, mass balance, momentum, and energy, as well as the equations of state and elastic-plastic Prandtl–Reuss flow:

- equations of trajectory of material particles motion

$$\dot{x}_i = u_i; \tag{1}$$

- equation of medium continuity

$$V_0 \rho_0 = V \rho; \tag{2}$$

- law of change of a material particle momentum

$$\rho \dot{u}_i = \sigma_{ij,j}; \tag{3}$$

- change of internal energy of a particle

$$\rho \dot{e} = \sigma_{ij} \dot{\epsilon}_{ij}; \quad (4)$$

- the strain rate tensor is of the form

$$\dot{\epsilon}_{ij} = 0.5(u_{i,j} + u_{j,i}); \quad (5)$$

- the stress tensor can be presented in the standard form

$$\sigma_{ij} = -\delta_{ij}P + s_{ij}, \quad (6)$$

where  $s_{ij}$  is deviator of the stress tensor, responsible for the reaction to shear deformation of the material particles;  $\delta_{ij}$  is Kronecker delta;  $P$  is function of pressure in the form of Mie–Grüneisen.

The process equations are taken in the Prandtl–Reuss form

$$\hat{s}_{ij} + d\lambda' s_{ij} = 2G\dot{\epsilon}'_{ij}, \quad \dot{\epsilon}'_{ij} = \dot{\epsilon}_{ij} - \dot{\epsilon}_{kk}/3 \quad (7)$$

provided Huber–Mises plasticity

$$s_{ij}s_{ij} \leq 2Y_0^2/3, \quad (8)$$

where  $Y_0$  is the dynamic yield strength, and scalar factor  $d\lambda'$  is determined by the known procedure of reducing to the yield circle. The above equations use the standard notation: each of indices  $i, j$  takes values of 1, 2 and 3, the repeated indices assume summation; dot over a symbol is the time derivative, index after comma is derivative by the corresponding coordinate,  $x_i, u_i$  are components of vectors of the position and velocity of a particle, respectively,  $\rho$  is the current density,  $G$  is the shear modulus.

### 3 EQUATION OF STATE

We consider a three-term equation of state with the solid-phase free energy being determined as

$$F(V, T) = E_x(V) + c_{v,l}T \ln \left( \frac{\Theta(V)}{T} \right) - \frac{1}{2}c_{v,e0}T^2 \left( \frac{V}{V_0} \right)^{2/3}, \quad (9)$$

where  $V$  is the specific volume,  $E_x(V)$  is the “cold” energy,  $T$  is the temperature,  $c_{v,l} = 3R/A$  is the specific heat of the lattice at constant volume,  $A$  is the mean atomic weight,  $R$  is the gas constant,  $\Theta(V)$  is the Debye temperature, and  $c_{v,e0}$  is the experimental value of the electron heat capacity under standard conditions. The elastic (cold) component of energy  $E_x(V)$  is related

exclusively to interaction forces between the body atoms and is equal (including the energy of zero vibrations) to the specific internal energy at the absolute zero temperature.

The thermodynamic model of a few-parameter equation of state is based on the dependence of the Grüneisen coefficient  $\gamma$  on the volume

$$\gamma(V) = \frac{2}{3} - \frac{2}{(1 - aV_0/V)}, \quad a = 1 + \frac{2}{(\gamma_s - 2/3)} + \frac{2P_{t,0}}{K_s}, \quad (10)$$

where  $\gamma_s = \beta K_s V_0 / c_v$ ;  $K_s$  is the adiabatic modulus of volume compression,  $c_v$  is the specific heat at constant volume, and  $P_{t,0}$  is the thermal pressure in the initial state.

The general expression for the volume dependence of the Grüneisen coefficient has the form

$$\gamma(V) = - \left( \frac{2-t}{3} \right) - \frac{V}{2} \frac{\frac{d^2}{dV^2} (P_x V^{2t/3})}{\frac{d}{dV} (P_x V^{2t/3})}. \quad (11)$$

In equation (11), the situation value corresponds to the Landau–Slater theory [17, 18] at  $t = 0$ , to the Dugdale–McDonald theory [19] at  $t = 1$ , and to the free-volume theory [20] at  $t = 2$ .

To determine the zero isotherm, we equated the expression for the Grüneisen coefficient (10) at the zero temperature  $T = 0$  K to the expression for the generalized Grüneisen coefficient (11)

$$\frac{2}{3} - \frac{2}{1 - a_x V_0/V} = - \left( \frac{2-t}{3} \right) - \frac{V}{2} \frac{\frac{d^2}{dV^2} (P_x V^{2t/3})}{\frac{d}{dV} (P_x V^{2t/3})}. \quad (12)$$

Here,  $a_x$  is the value of the parameter  $a|_{T=0}$  at the zero temperature in equation (10), which can be taken  $a_x = a(0) = 1 + 2/(\gamma_s - 2/3)$  as the first approximation.

#### 4 GENERATION OF THE DIFFERENCE GRID

The computational domains for a complex technical object are a set of simple and multiply connected spatial figures to be filled with difference grid. The spatial triangulation is the most versatile method of constructing computational grids in multiply connected domains of arbitrary shape. Following the successful methodology for constructing triangular grids for two-dimensional case in multiply connected domains described in [21,22], we extend it to the spatial case. For this, it is necessary to implement the following steps:

1. to construct geometric primitives for the boundary of complex multiply connected technical objects;

2. to select the majorizing volume and, then, fill it with the mesh nodes;
3. to import the triangular grid on the boundary surfaces;
4. to remove nodes of the original grid that do not fall in the predetermined volume of the multiply connected object;
5. to apply the dynamic method to the resulting set of both boundary and internal nodes for their placement into an ordered structure;
6. to triangulate the spatial set of nodes and form the structure of logically connected difference grid for the numerical solution of dynamic problems of deformation and fracture;
7. to evaluate the resulting spatial mesh by the quality criteria.

Without going into detail on all counts, consider the dynamic method successfully applied in [16]. Since the computational domain, which is a multiply connected spatial object, is filled with the primary grid, it remains to apply the dynamic method to the resulting set of both boundary and internal nodes for their placement into an ordered structure. The physical basis of the dynamic method is interaction of the nearby nodes (particles). Therefore, the type of the force is chosen so that the whole system of particles in equilibrium provides a dense packing or a touch (small overlap) of spheres of the nodes influence. The cell size is determined by the distance between the centers of adjacent nodes-particles. The form of the interaction force must be finish in the space, i.e., the repulsive (attractive) force acts when the two nodes are at a distance nearer (farther) than the equilibrium distance. When the nodes are removed at a considerable distance from each other, they cease to interact. The form of the force of interaction between the particles is chosen to be like the force of van der Waals [8, 16]. Note that for this kind of interparticle force, all the “bad” elements are physically unstable, and, therefore, the chance of creation of such “bad” cells is significantly reduced.

Based on the initial arrangement of the particles, the differential-difference equation is integrated with respect to time, until the system reaches equilibrium. The mathematical formulation of the movement equation for  $i$ -th node in the vector form is

$$m_i \frac{d^2 \vec{x}_i(t)}{dt^2} + K_i \frac{d \vec{x}_i(t)}{dt} = \vec{f}_i(t),$$

where  $\vec{x}_i$  is position of  $i$ -th node-particle in the computational domain,  $m_i$  is mass of the node-particle;  $K_i$  is damping factor,

$$\vec{f}_i = \sum_{j=1}^{j=N} \vec{f}_{ij}$$

is force acting on  $i$ -th node,  $N$  is number of neighbors of  $i$ -th particle,

$$\vec{f}_{ij} = \vec{f}(r_{ij})$$

is vector of force of interaction between nodes  $i$  and  $j$ ,

$$r_{ij} = ((x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2)^{1/2}$$

is distance between nodes  $i$  and  $j$ .

Solving the resulting dissipative problem for the establishment, we obtain a system of nodes that have come to equilibrium. To successfully solve the space problem, one needs to be very careful when choosing the interaction forces between the nodes. Even though the flat case allows for single-particle interaction potential of Lennard-Jones type, the three-dimensional case requires the use of a many-particle potential to ensure the logical connection (preserving the angles) between adjacent grid points during movement of the mesh nodes. Since we have not found the interaction force providing the ideal geometric shape of a tetrahedron, we impose the logical connection to the grid nodes using Delaunay triangulation.

The final stage of the work is global optimization. The necessity of this procedure is due to the possible simplifications of the generation algorithm or the ambiguity of basic quality criterion. For example, constructing tetrahedral mesh corresponding to the Delaunay criterion may result in possible presence of degenerate tetrahedra whose vertices are on the surface of the same sphere near its equator. The basic method of optimization consists of local retriangulation of polyhedrons built on one of the fixed combinations of mesh elements. In three dimensions, it is usually a question of decomposition of a hexagon or octagon, but more elaborate options are possible too. The main distinctive feature of the algorithms built according to this principle is the formalization of reconstruction procedure and criteria for evaluating alternative options for decomposition. For example, in well-known work [23], the base grid primitive is a triangular face. The face remains in the grid if none of fixed transformations of polyhedra containing it improves the minimum value of quality criteria of the corresponding combination of tetrahedra.

Another method of improving the grid quality is local transformations [24, 25]. At each step of the algorithm, all tetrahedra are considered in turn, transformations are performed in the cases where it is advantageous. The process is repeated until the algorithm stop condition is fulfilled. Advantage of this method is that at each step the mesh quality improves monotonously because we make the transformation only when it improves the quality. At the same time it is its disadvantage: the method can get “stuck” in a local optimum and do not come close to the globally optimal grid.

For the whole mesh, an ordered set of qualities of all tetrahedra is used. The sets are com-

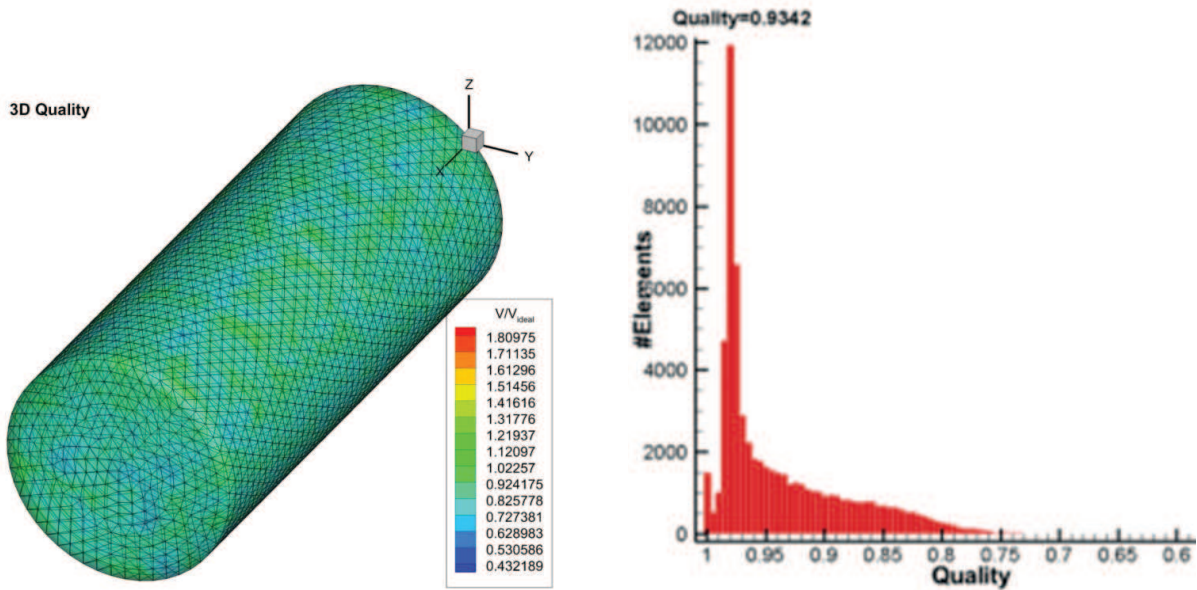


Figure 1: Quality of 3D mesh for a cylinder.

pared lexicographically (this means that of two grids the one that has better quality of the worst tetrahedron is better; if the qualities of the worst tetrahedra of two grids are the same, then, the second tetrahedra from the end of the quality are compared, etc.). In practice, we never need to build an ordered array of qualities of all tetrahedra.

The quality of an individual grid element (cell) is usually estimated by calculating the characteristic parameter  $Q$ , which value depends on the shape of the tetrahedron, followed by comparing the result with the value of a similar parameter from a regular tetrahedron.

For consistency, the value of the quality criterion is converted so that the normalized value of  $Q$  is in the range  $(0, 1]$ . At that,  $Q = 1$  corresponds to a regular tetrahedron. It should be noted that the generally accepted quality criteria of tetrahedra only conditionally reflect the degree of proximity to the element of regular shape and are not equivalent to each other. The simplest example of quality is the ratio of radius of the circumscribed sphere to radius of the inscribed one:

$$\beta = \frac{R_c}{3R_i}.$$

To demonstrate the operation of the automatic mesh generator, we present the result of the mesh generation for a simple 3D shape: cylinder (the radius is 1, the length is 5). Figure 1 shows the 3D mesh of this cylinder, and the color distribution reflects the ratio of the tetrahedra volumes. As one can see from the figure 1, the majority of the tetrahedra have volume close to the ideal one. The average quality of the mesh of the built cylinder is equal to 0.93, and the quality of the worst element of the mesh  $Q_{\min} = 0.73$ . OX axis shows the quality in the reverse order, from

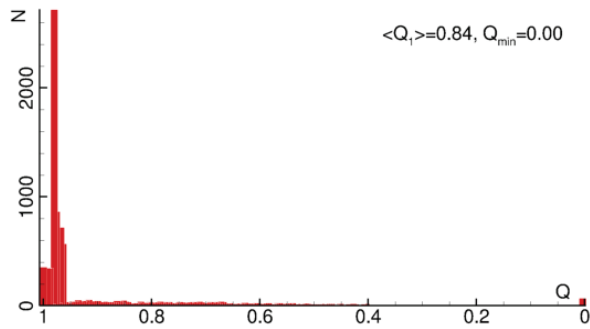


Figure 2: NETGEN. Quality of 3D tetrahedral mesh.

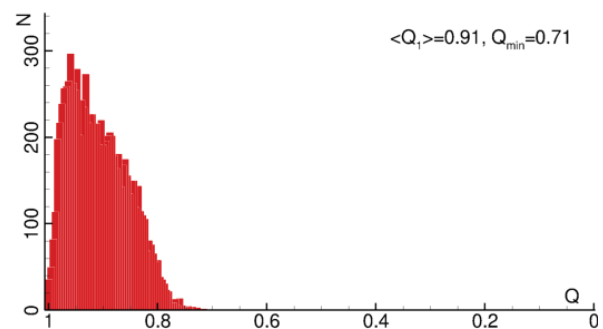


Figure 3: NETGEN. Quality of 3D tetrahedral mesh after improvement.

the best to the worst, OY axis shows the number of elements.

To compare the proposed mesh generator with existing analogues let us consider third-party projects. As the first example, consider SALOME project [26] that uses freeware generator NETGEN [27] for meshing. NETGEN is an automatic mesh generator for two-dimensional and three-dimensional domains. The program generates tetrahedral meshes in a three-dimensional domain.

Figure 2 shows the original quality of a mesh built in SALOME. The histogram has a clear line spectrum in two or three regions and a large spread of cells with a “bad” quality in regions near the border, reaching the values close to zero. Even though the average quality of the mesh is acceptable,  $\langle Q_1 \rangle$  shows 0.84, the quality of the “worst” element  $Q_{\min}$  is close to zero, which makes it impossible to use this grid to calculate high-speed loading of solids. This means that most of the tetrahedra created by the the mesh generator are close to the ideal one (about 90%), while the other ones are formed “as is”, only with logical connections to the mesh as a whole. The result of the algorithm presented above and implemented in the software package is shown in figure 3. Application of the dynamic method significantly extends the number of quality cells, while the cells with low quality “disappear”, because in the process of dynamic correction their quality improves. The figure shows how the average quality of the grid improved from  $\langle Q_1 \rangle = 0.84$  to  $\langle Q_1 \rangle = 0.91$ . Application of the mesh local restructuring, after the dynamic impact, significantly improves the histogram, and the worst mesh element has quality  $Q_{\min} = 0.71$  instead of almost zero initially.

The second example demonstrating the algorithms for improving quality and uniformity of the difference grid is commercial package ANSYS [28], which constructs grids using a built-in generator. Figure 4 shows the initial quality of a mesh built in ANSYS. Despite the fact that the average quality by all the criteria is lower than that in SALOME generator, the quality histogram is not line one, and the spectrum tends to a normal distribution. It is very important that the number of bad cells is small, and a built-in mesh generator does not create degraded



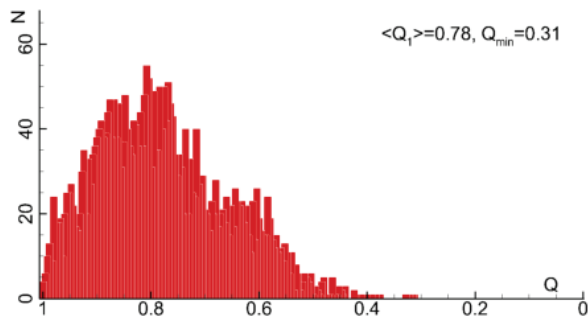


Figure 4: ANSYS. Quality of 3D tetrahedral mesh.

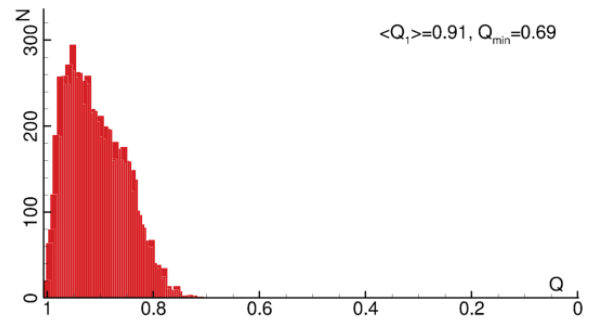


Figure 5: ANSYS. Quality of 3D tetrahedral mesh after improvement.

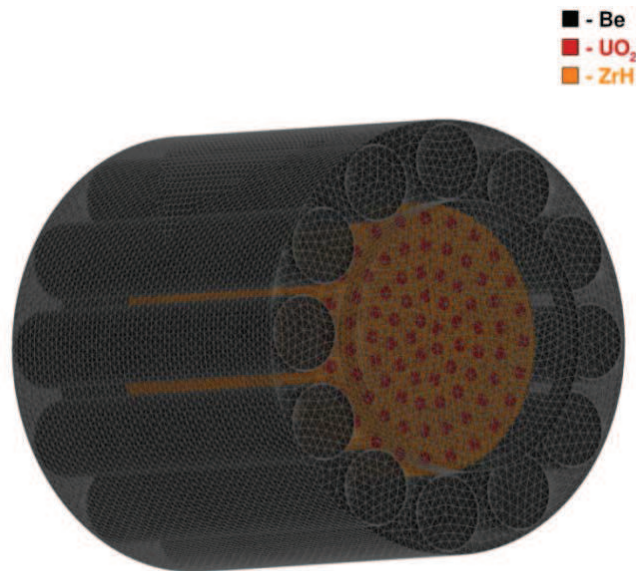


Figure 6: Building 3D mesh in a model reactor presented by a set of elementary and multiply connected figures.

tetrahedra. This grid can be used in the simulation of high-speed deformation, but its quality can be significantly improved. Figure 5 shows the result of the quality improvement algorithms. One can see from the figure that the quality of the mesh  $\langle Q_1 \rangle$  is significantly improved, the spectrum center has shifted to higher values of quality, and the quality of the “worst” tetrahedron  $Q_{\min}$  has increased more than twice.

## 5 EXAMPLES OF APPLICATION OF THE TOOL FOR HIGH-VELOCITY INTERACTION AND DAMAGE OF SOLIDS

To illustrate the efficiency of the numerical 3D tools we consider the interaction of a model reactor of thermionic power plant with 12 regulating cylinders in side reflector of neutrons and

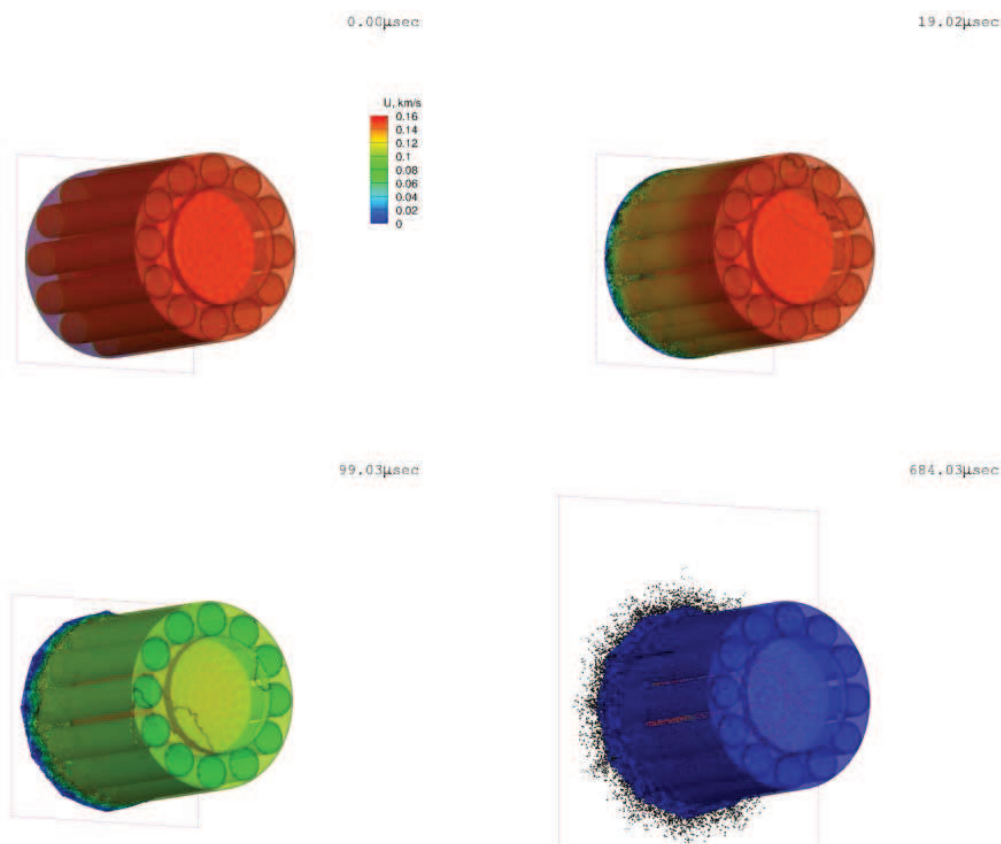


Figure 7: Frame sequence of a process of model reactor collision with a hard surface.

concentric rows of power-generating channels in the moderator of zirconium hydride. EGC are fuel elements with integrated thermionic converter of thermal energy of nuclear reaction to electric energy. The large number of contact surfaces and various materials show that the software package allows one to solve the problem to the “end”, i.e., until a complete failure and shutdown of the model reactor fragments. Figure 6 is an illustration of the geometric model of the reactor.

Consider the simulation of the reactor impact in a 3D setting (figure 7). The typical values—impact velocity is 200 m/s, the reactor contains 1.5 million computational cells, the process time  $t = 685 \mu\text{s}$  was achieved in 150 hours of CPU time. During the contact, under the action of strong compressive stress, a set of material deformations takes place near the contact surface. Upon reaching the critical strain value, the reactor shell of beryllium starts to be destroyed (figure 7 and 8 at moment  $t = 19 \mu\text{s}$ ). By time  $t = 100 \mu\text{s}$ , the lower cover is virtually destroyed and the “filling” of zirconium hydride begins to break down. Further, under the action of the tensile stresses arising from the numerous interference waves, starts the process of destruction of reactor core, both hydride of zirconium and uranium dioxide. The process of collision ends

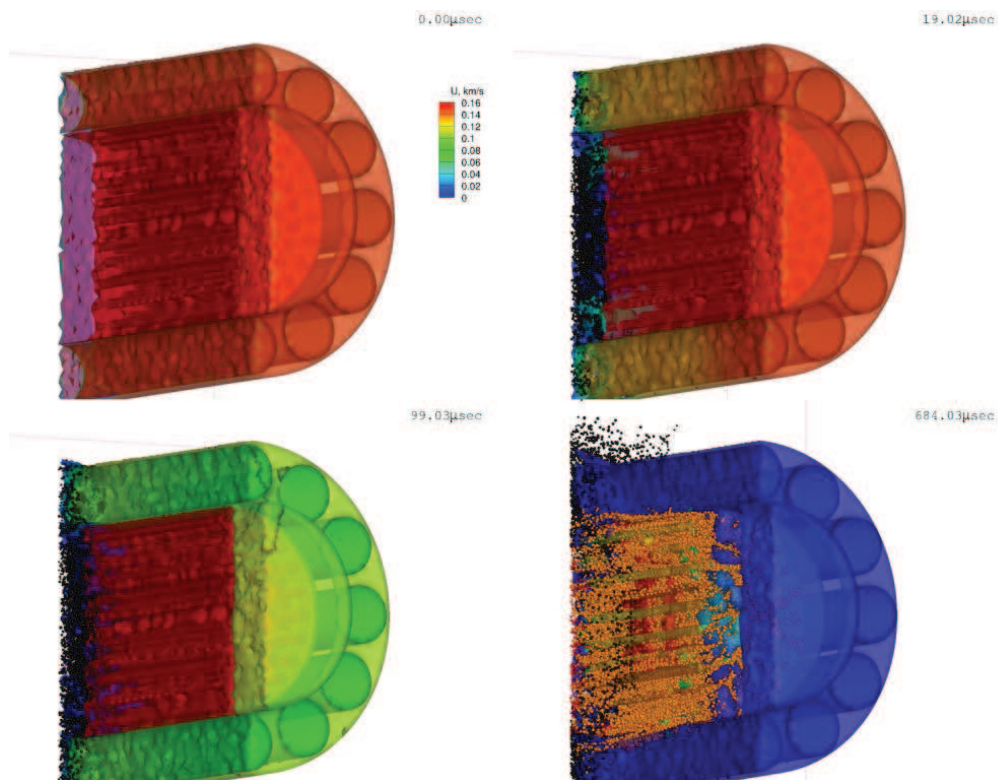


Figure 8: Frame sequence of a process of model reactor collision with a hard surface (the reactor section is shown to illustrate the destruction of the inner content).

with the destruction of the bottom part of the structure shell and almost complete destruction of the “filling” of zirconium hydride. At that, the fuel elements are fractured only at the contact boundary (figure 7 at moment  $t = 685 \mu s$ ).

Externally, the destruction is revealed in a small layer of tiny fragments (figure 7, at moment  $t = 100 \mu s$ ) and a halo of flying fragments at moment  $t = 685 \mu s$ .

## 6 CONCLUSIONS

It is shown that the existing software package REACTOR3D has evolved and is suitable to solve the impact problems in the 3D case. The method of dynamic construction of tetrahedral mesh for complex, multiply connected objects is presented. The algorithm for modeling fractured material by discrete particles of finite size is exposed. The key moments of interaction of the fractured material fragments with the boundaries of solid material and with each other are shown. The examples of efficiency of the computational modules for both the regions of solid material and numerous fragments of the fractured material are given.

**Acknowledgments:** The work is supported by grant of the Russian Foundation for Basic Research (No. 16-08-00244).

The paper is based on the proceedings of the XXXII International Conference on Interaction of Intense Energy Fluxes with Matter, which was held in Elbrus settlement, in the Kabardino-Balkar Republic of the Russian Federation, during March 1–6, 2017.

## REFERENCES

- [1] V. E. Fortov, V. V. Kim, I. V. Lomonosov, A. V. Matveichev, and A. V. Ostriuk, *Int. J. Impact Eng.* **33**(1–12), 244–253 (2006).
- [2] M. E. Povarnitsyn, K. V. Khishchenko, and P. R. Levashov, *Int. J. Impact Eng.* **33**(1–12), 625–633 (2006).
- [3] E. I. Kraus and I. I. Shabalin, *Frattura ed Integrità Strutturale* **24**, 138 (2013).
- [4] A. E. Mayer, K. V. Khishchenko, P. R. Levashov, and P. N. Mayer, *J. Appl. Phys.* **113**(19), 193508 (2013).
- [5] A. V. Radchenko, P. A. Radchenko, and S. P. Batuev, *Russ. Phys. J.* **58**, 319 (2015).
- [6] A. E. Buzyurkin, I. L. Gladky, and E. I. Kraus, *Aerosp. Sci. Technol.* **45**, 121 (2015).
- [7] P. A. Radchenko, S. P. Batuev, A. V. Radchenko, and A. M. Tukaev, Simulation of interaction of damage agents of different shape with shaped-charge munition, in: *AIP Conf. Proc.* **1800** (2017).
- [8] E. I. Kraus and I. I. Shabalin, Reactor2D: A tool for simulation of shock deformation, in: *AIP Conf. Proc.* **1770** (2016).
- [9] M. L. Wilkins, *Computer Simulation of Dynamic Phenomena*, Scientific Computation (Springer Berlin Heidelberg, Berlin, Heidelberg, 1999).
- [10] V. M. Fomin, A. I. Gulidov, and G. A. Sapozhnikov, *High-Velocity Solids Interaction* (SB RAS, Novosibirsk, 1999), in Russian.
- [11] V. M. Fomin, E. I. Kraus, and I. I. Shabalin, *Mater. Phys. Mech.* **7**, 23 (2004).
- [12] E. I. Kraus, V. M. Fomin, and I. I. Shabalin, *Bulletin of the South Ural State University, Series: Mathematical Modelling, Programming and Computer Software* **7**(1) (2014).
- [13] E. I. Kraus and I. I. Shabalin, *J. Phys.: Conf. Ser.* **774**, 012009 (2016).
- [14] V. M. Fomin, E. I. Kraus, and I. I. Shabalin, A few-parameter equation of state of the condensed matter and its application to the impact problems, in: *EPJ Web Conf.* **10** (2011).
- [15] E. I. Kraus and I. I. Shabalin, *J. Phys.: Conf. Ser.* **653**, 012085 (2015).
- [16] K. Shimada and D. C. Gossard, *Proceedings of the third ACM Symposium on Solid Modeling and Applications (Proc. SMA '95)* pp. 409–419 (1995).
- [17] I. C. Slater, *Introduction in the Chemical Physics* (McGraw Book company, Inc., New York London, 1935).
- [18] L. D. Landau and K. P. Stanyukovich, *Dokl. Akad. Nauk SSSR* **46**, 399–403 (1945).
- [19] J. S. Dugdale and D. McDonald, *Phys. Rev.* **89**, 832–851 (1953).
- [20] V. Ya. Vaschenko and V. N. Zubarev, *Sov. Phys. Solid State* **5**, 653–655 (1963).
- [21] E. I. Kraus, V. M. Fomin, and I. I. Shabalin, *Vych. Tekhn.* **11**, 104 (2006).
- [22] E. I. Kraus, V. M. Fomin, and I. I. Shabalin, *Vych. Tekhn.* **14**, 40 (2009).
- [23] B. Joe, *SIAM Journal on Scientific Computing* **16**(6), 1292–1307 (1995).
- [24] B. M. Klingner and J. R. Shewchuk, Aggressive Tetrahedral Mesh Improvement, in: *Proceedings of the 16th International Meshing Roundtable* (Springer Berlin Heidelberg, Berlin, Heidelberg, 2008), pp. 3–23.
- [25] L. A. Freitag and C. Ollivier-Gooch, *International Journal for Numerical Methods in Engineering* **40**(21), 3979–4002 (1997).
- [26] SALOME, URL: <http://www.salome-platform.org/>.
- [27] NGSolve, Netgen mesh generator, URL: <https://ngsolve.org/>.
- [28] ANSYS, URL: <http://www.ansys.com/>.

Received June 13, 2017