RAMZES-CP Technique for Solving of Spatial Multimaterial Heat Conduction Flows in Eulerian-Lagrangian Coordinates.

A.N. Bykov, V.A. Veselov, B.L. Voronin, A.M. Erofeev

Russian Federal Nuclear Center – VNIIEF, 37 Mir Ave., Sarov, N.-Novgorod region, Russia

The paper discusses technique RAMZES-KP designed for computing spatial multi-component heat-conducting flows in Eulerian-Lagrangian coordinates on parallel distributed-memory computer systems.

Introduction

Technique RAMZES-KP /1/ is designed for computing multi-component heat-conducting flows in Eulerian-Lagrangian coordinates on parallel distributed-memory computer systems.

The technique is based on the following principles:

- using the gas dynamics equations and heat conduction equation written both in the Cartesian and curvilinear coordinate systems, in Eulerian-Lagrangian variables;
- using the finite difference approximation in time both to the heat conduction equation and the gas dynamics equations;
- splitting into physical processes;
- partition of the problem geometry to fragments, in each of which its most appropriate grid is constructed;
- the resultant system of multidimensional finite difference equations is solved using the splitting into directions and the direct method for solution of finite difference equation subsystem on block-matrix type grids, i.e. the integrated sweep method, which has been developed by the authors;

- the method of concentrations and Yangs method are used for construction of interfaces in the computations of material flows from mixed cells.

Also, a feature of the technique is that parallel computations are employed at all phases (preprocessing, computation, and analysis of results) of the task run on the multi-processor distributed-memory computer.

Governing equations and the method for their solution

The technique RAMZES-KP solves the following gas-dynamics equation system for multi-component medium:

$$\frac{d\rho_i}{dt} = -\rho_i \cdot div\bar{U}, \quad i = 1,...,N$$

$$\frac{d\bar{U}}{dt} = -\frac{1}{\rho} \cdot gradP$$

$$\frac{d\varepsilon_i}{dt} = -\frac{P_i}{\rho_i} \cdot div\bar{U}, \quad i = 1,...,N$$

$$P_i = P_i(\rho_i, \varepsilon_i), \quad i = 1,...,N$$

Here ***N*** is the amount of materials in the mixture. To close the equation system, additional assumptions of the materials in the mixture should be involved.

$$\rho = \sum_{i=1}^{N} \rho_i \cdot \xi_i$$

$$\frac{d\xi_i}{dt} = 0, \quad i = 1,...,N$$

$$P = \sum_{i=1}^{N} P_i(\rho_i, \varepsilon_i) \cdot \xi_i$$

If equal pressures of the components are assumed, the volume concentrations and pressures should be changed according to the condition of equal pressures of the components:

$$P = P_i\left(\rho_i(\xi_i), \varepsilon_i(\xi_i)\right)$$

For multi-component medium the heat conduction equation system is solved in the following form:

$$\frac{d\varepsilon}{dt} = \frac{1}{\rho} div(\chi \cdot gradT)$$
$$\varepsilon_i = \varepsilon_i(\rho_i, T), i = 1,...,N$$
$$\chi_i = \chi_i(\rho_i, T), i = 1,...,N$$

To close the system, additional assumptions of the materials in the mixture should be involved.

$$\varepsilon = \sum_{i=1}^{N} \varepsilon_i \cdot \eta_i$$

$$\chi = \frac{\prod_{i=1}^{N} \Lambda_i}{\sum_{i=1}^{N} \left(\xi_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{N} \Lambda_j \right)} \cdot \sum_{i=1}^{N} \Phi_i \xi_i$$

Here: $\frac{\boldsymbol{d}}{\boldsymbol{dt}}$ is the total time derivative,

$\boldsymbol{\rho}$ is the cell density,

ρ_i is density of components in the mixed cell,

$\boldsymbol{P}$ is the cell pressure,

$\bar{\boldsymbol{U}} = \bar{\boldsymbol{U}}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w})$ is the mass velocity vector,

ε is specific internal energy,

ε_i is specific internal energy of components in the mixed cell,

T is the cell temperature,

χ is the cell heat conductivity factor,

$\chi_i = \Lambda_i$*Ш$_i$ is the heat conductivity factor of components in the mixed cell,

ξ_i is the volume concentration of components in the mixed cell,

η_i is the mass concentration of components in the mixed cell,

$\Phi_i = \frac{\partial \varepsilon_i}{\partial T}$.

The specific form of differential operators *div* and *grad* depends on the coordinate system, which the mass velocity vector decomposition is performed in.

The technique RAMZES-KP employs the curvilinear coordinate system (r, S, φ) including, as a special case, the spherical, cylindrical and toroidal coordinate systems (see Fig. 1).

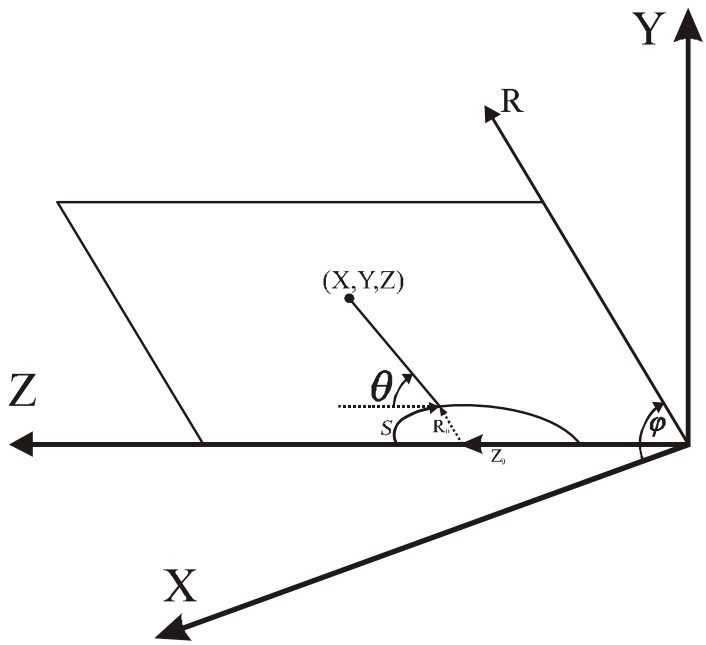

Fig. 1- Curvilinear coordinate system

The relation between the Cartesian and curvilinear coordinates is given as

$$X = (r \cdot \sin\theta(S) + R_0(S)) \cdot \cos\varphi$$
$$Y = (r \cdot \sin\theta(S) + R_0(S)) \cdot \sin\varphi$$
$$Z = r \cdot \cos\theta(S) + Z_0(S)$$

Coordinates S and φ are Eulerian and coordinate r is originally Lagrangian, but automatically, by some criteria, there can be a local refusal of the Lagrangian feature irrespective of the computation execution, that is the radial coordinate r is movable, with the law of its motion being able to vary in a wide range from the Lagrangian to the Eulerian.

The problem discretization is of a two-level nature: depending on the problem geometry the whole geometry can be partitioned into fragments, each of which is described in its most suitable coordinate system (r, S, φ) given using a reference line.

The computational curvilinear grid is produced by intersection of surfaces S=const, φ=const, and r=const, where the family r=const typically includes interfaces of physical regions, with the family φ=const being unique for all the problem fragments and S=const being its own in each fragment. Thus, the computational grid is composed of curvilinear cells with faces S=const, φ=const, and r=const (Fig. 2) and, structurally, the grid is of the block-matrix form, with each block of the matrix corresponding to one of the problem fragments.

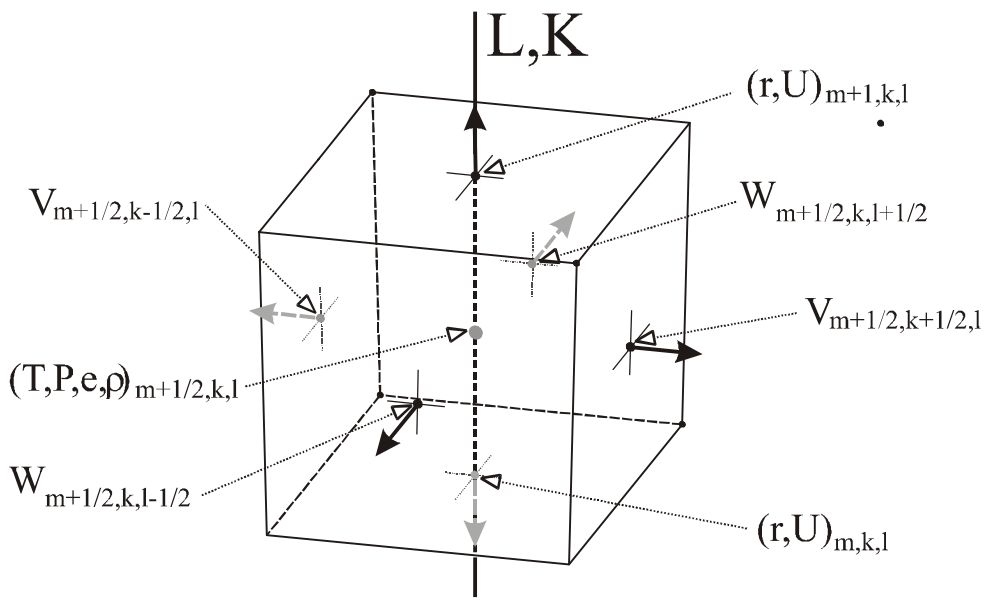

Fig. 2 - Centering of quantities

The spatial finite difference approximation to differential operators is performed on these block-matrix grids with account for centering of thermodynamic (ρ, T, ε, P) and cinematic ($\vec{r}$, $\vec{U}$) quantities as shown in Fig. 2.

In time, the implicit finite difference approximation both to the heat conduction equation and the gas-dynamic equations is used.

The resultant system of the multidimensional finite difference equations is solved using a method of splitting into directions. The method reduces the solution of the general linear algebraic equation system to a sequential solution of subsystems of these equations along each spatial direction. As the computational grids along each direction

are not purely Lagrangian (they are Eulerian along directions S and φ and movable along direction r), the solution reduces to two stages: Lagrangian and Eulerian.

For the Lagrangian stage, the structure of the matrix of these equation subsystems is either tridiagonal or close to tridiagonal. The solution of these equation subsystems is sought with the sweep method.

The Eulerian stage has been implemented in programs of interpolation of the grid along radius through superposition of the grids (before and after the interpolation), while in the computation of the non-Lagrangian motion of the grid along the radial direction and Eulerian directions S and φ the donor scheme of the first-order flow approximation is used. The mixture component flows from mixed cells are calculated with Yangs method /2/.

Parallel computations

A characteristic feature of the technique is using parallel computations at all stages (preprocessing, computation, and computed data analysis) of the problem run on the multi-processor distributed-memory computer.

The features of the parallelization methods are the following:

- the method should allow the problem solution on any number of processors, from 1 to P, where P is an arbitrary number in the general case;
- each processor stores data only on those points, which are being processed on this processor;
- the methods should allow arithmetic and communication operations at a time (given appropriate hardware on the computer);
- independence of the computation results on the number of the processors used.

For the gas dynamics and heat conduction problems it seems most natural to use the principle of geometrical parallelism in the problem solution domain decomposition over processors. The technique implements the capability of the problem partition into computational domains with subsequent decomposition of each computational domain to parallelepipeds in indexed space. Thus, the interprocessor communication proceeds both between and inside the domains. Three principal types of message passing between

processors have been developed that are used both in gas dynamics and heat conduction, i.e. parallel-pipeline method, communication of edges, communication between domains.

When developing the parallelization methods, timestep-invariable decomposition of the data matrix to submatrices was used. Alongside its merits, for example, locality of most communications, the method has its difficulties relating to arrangement of the parallel execution of a large amount of recurrent formulas of the sweep on the processor line. To overcome this difficulty, our own version of the parallel-pipeline method has been developed. The principal idea of the algorithm is that, having calculated sweep factors for the i–th channel and transmitted them to the line-next processor, each processor transfers to processing of the other channels. As the backward sweep arrives the processor interrupts its operation in the forward sweep computation and calculates the solution for the i–th channel. To optimize the communication operations, the sweep factors are passed between processors when a portion of the channels has been processed. The number of the channels in one portion is variable (from 1 to the total number of the channels) and changes automatically from step to step by the criterion of step computation time minimization. If the arithmetic and communication operations are executed by different devices, then the arithmetic and communication operations can be made concurrent. Fig. 3 shows several phases of the parallel-pipeline method for a line of four processors.

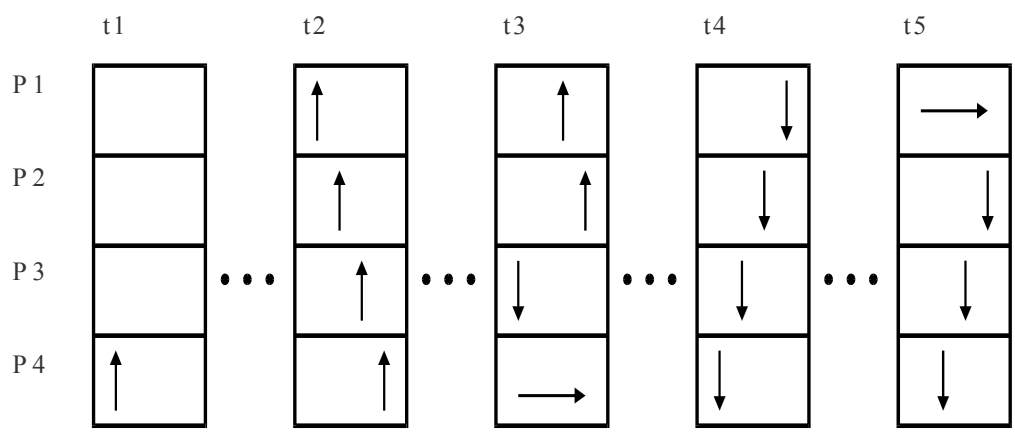

Fig. 3 – Several phases of the parallel-pipeline method

The parallelization methods developed provide an acceptable efficiency (50-60% with the step calculation time of the order of a few seconds, with a longer step time the efficiency being up to 60-80%) on the modern multiprocessor computers.

The Modern Directions of the Technique Development

The parallel computations revealed the problem of the crash feature of the technique. Whereas in the one-processor mode the crash was once in 24 hours, in the parallel mode that time was seconds when computing on several tens of processors. The reasons for the crashes have been analyzed and two principal directions of the refinement of the technique have been selected to reduce its crash feature:

- using combined grids,
- local refusal of the Lagrangian feature in the computation on the Eulerian-Lagrangian meshes.

Below is the brief description of the works on each of these directions.

Either moving (Lagrangian) or unmoving (Eulerian) regular computational grids can be used for the gas dynamics problem solution.

Even in computation of one 2D difference grid direction the Lagrangian approximation makes the computation cost-efficient in terms of the number of needed grid nodes and computation amount; the computation results are highly accurate; material interfaces are tracked automatically. However, when using moving grids, there can be cell overlaps leading to crashes; and in computations of severe vortex flows the requirement of simple connectivity of the region calculated on the regular grid can be violated.

The Eulerian computational grids are free of these moving grid related disadvantages, but exhibit a number of special features. The principal special feature is the need for tracking material interfaces not connected to the grid. Besides, when using the Lagrangian grid, the region under consideration is always approximated by the same number of the grid nodes and, hence, the initial accuracy of the approximation is such in the complete computation as well. In the case of the Eulerian grids the approximation accuracy depends on the grid spacing, for the results to be accurate enough the number of the cells should be fairly large (be higher by several times than the number of the Lagrangian grid cells).

Clear that neither purely Lagrangian nor purely Eulerian computation method is ideally suitable for many problems, while the best computational method can be some combination of the Eulerian and Lagrangian approximations.

Experience suggests that in most problems the computational domain can be divided into subdomains, so that for large time intervals each subdomain be best approximated by the moving and unmoving grids. The idea to use the combination of the Eulerian and Lagrangian approximation was proposed by V.F. Noh in 1964 /3/. We employ orthogonal rectangular Eulerian grids and moving regular Eulerian-Lagrangian meshes. We term the computation using these grids as the computation on combined grids. Unmoving orthogonal cylindrical grids are used either in those parts of the problem, where by virtue of geometry and (or) irregular nature of gas-dynamic flows the regular moving computational grid is hard to construct and maintain during the numerical solution of the problem or in those parts of the problem, where the moving grids have a singularity, for example, a center.

Consider the algorithms of the computation on combined grids by the example of the problem of strong explosion /4/. At the initial time in a region of radius 1 density of 100 and specific internal energy of 1000 are given; outside the sphere of radius 1 density is 1 and specific internal energy is 0.001. The equation of state in each region is ideal gas. A diverging shock wave is produced in medium under the action of pressure gradient.

The internal region will be approximated by the moving Lagrangian spherical grid (fragment No.1), the external region by the unmoving cylindrical Eulerian mesh (fragment No.2) (Fig. 4).

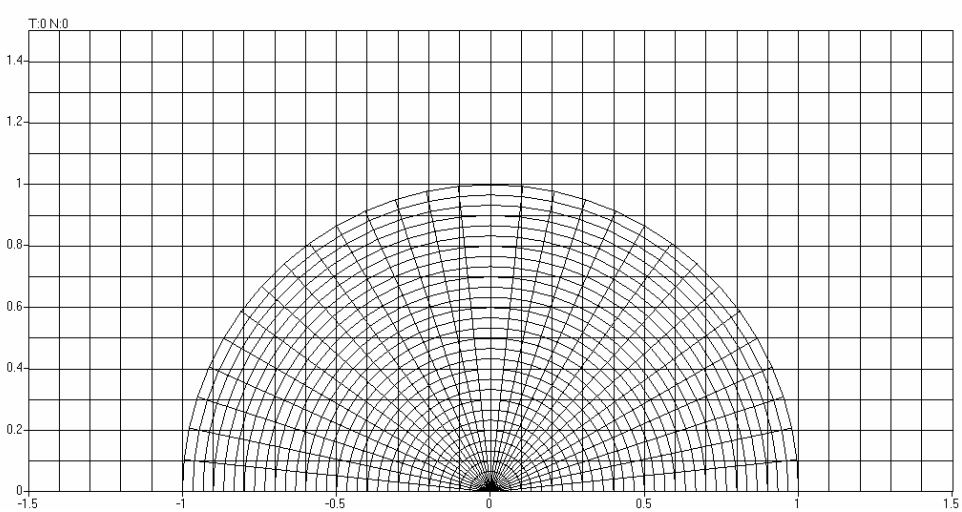

Fig. 4 - The computational grid at the initial time

At the task start the volume of Lagrangian material (the material from the Lagrangian grid) is estimated by recalculation for each Eulerian mesh cell. The Eulerian grid cells are distributed over groups (Fig. 5):

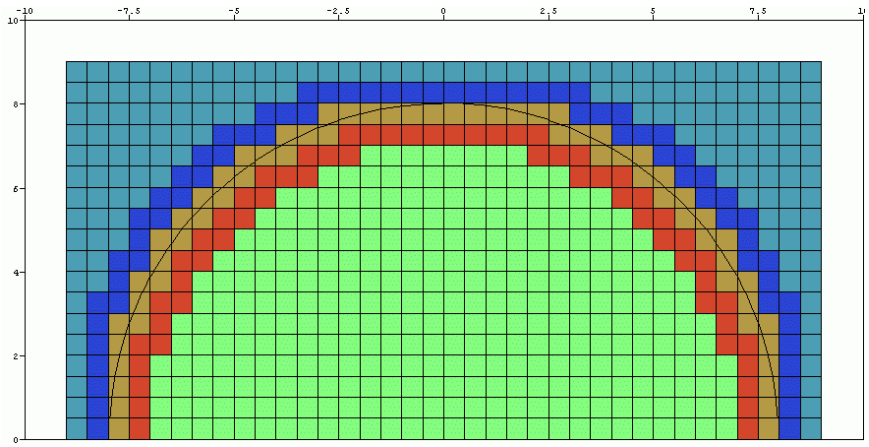

Fig. 5 –Eulerian grid cell distribution over groups

1. *Calculated* cell (light-blue). It is filled completely with the Eulerian material.
2. *Near-boundary calculated* cell (blue). It is also filled completely with the Eulerian material and is a neighbor of the *boundary* cell.
3. *Boundary* cell (brown). It is a mixed cell that contains two materials or is filled completely with the Eulerian material, borders on the pure cell filled with the Lagrangian material.
4. *Near-boundary non-calculated* cell (red). It is filled completely with the Lagrangian material and is a neighbor of the *boundary* cell.
5. *Non-calculated* cell (green). It is filled completely with the Lagrangian material, is not involved in the computations.

Thus, the moving boundary in the Eulerian fragment is described by a continuous boundary cell layer (tangency through the cell angle is possible) surrounded with layers of near-boundary calculated and non-calculated cells (here the tangencies are possible only through cell edges).

This distribution is calculated once at the start by recalculation over all Eulerian grid cells. During the computation the recalculations are performed only over the boundary and near-boundary cells, the cells can transfer from one type to another, with the layered structure of the layers being maintained additionally (the continuous layer of

the boundary cells is surrounded with the layers of the calculated and non-calculated cells).

When developing the algorithms for communication of boundary conditions between the moving and unmoving fragments, it was taken into consideration that the boundary condition passing should be before the step computation; the fragments should be computed by standard computer programs independently. Later it will allow using multi-processor distributed-memory computers for the computation on the combined grids.

In the first version of the fragment communication algorithm, pressures and velocities are passed from the Lagrangian fragment to the Eulerian fragment using the grid superposition by the beam-scanning method, with the moving boundary of the Lagrangian fragment, which is projected onto the Eulerian fragment, being the material interface. Flow of the materials through the boundary is impossible. The recalculation is performed for the boundary and near-boundary cells of the Eulerian fragment. All other needed components, i.e. density, energy, temperature, pressure, and two momentum components, are also recalculated in computing the Lagrangian material volume in the Eulerian grid cell. These quantities are entered into the relevant Eulerian grid arrays. The recalculated Lagrangian material momentum components are used to update the velocity components on the Eulerian mesh. This is done at the beginning of each timestep. After that the gas-dynamic process is calculated with a standard program. When calculating the new pressure field, the calculability scale is used extensively: for the non-calculated and near-boundary non-calculated cells the values of the calculability parameter are negative, which means that the sweep factors and new pressures are not calculated; the pressure proves imposed from the Lagrangian fragment. For all the Eulerian-material cells (calculated, near-boundary calculated and boundary cells) the complete gas dynamics computation is performed.

In the first version of the fragment communication algorithm, pressure is imposed on the Lagrangian fragment boundary from the Eulerian fragment. This is done as follows (Fig. 3). The Lagrangian boundary is represented as a broken line, pressure should be imposed on each segment of the line. Intersections of the broken line with all Eulerian grid boundary cells are sought. The cell contribution is determined by the length

of the intersection straight-line segment and then the desired pressure is sought by averaging. For example (Fig. 4), for straight-line segment AB: $P = \frac{P1 * AC + P2 * CD + P3 * DE + P4 * EB}{AB}$. Material pressure of the Eulerian component is taken for the cell pressures.

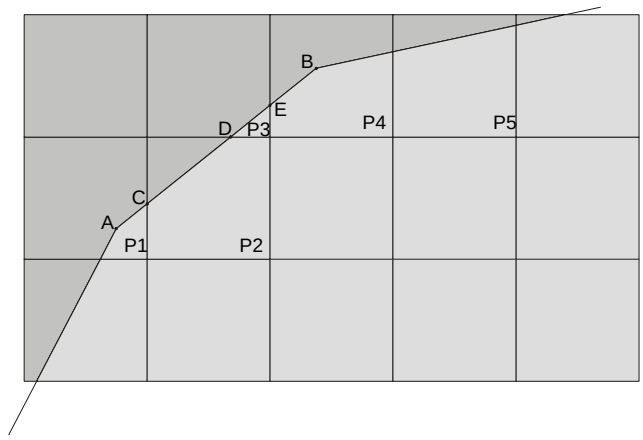

Fig. 6 – Determination of pressures imposed on the Lagrangian fragment

In the second version, a modified scheme of the boundary condition passing has been suggested, when the Lagrangian fragment contains a layer of dummy cells, in which the grid quantities will be computed by recalculation from the Eulerian grid. The following algorithm, which is simpler in implementation, has been proposed and tried:

- In the Eulerian-Lagrangian fragment, add a dummy domain covering the computational cells of the Eulerian grid. In that domain EOS corresponds to EOS of the Eulerian grid material.
- Impose velocity determined by interpolation over the Eulerian fragment cells on the boundary of the dummy domain.
- The scheme of the passing of pressures and velocities to the Eulerian fragment remains unchanged.

Results of the problem computation.

Fig. 7 presents the grid at the initial time. Fig. 8 shows the problem state at one of the times.

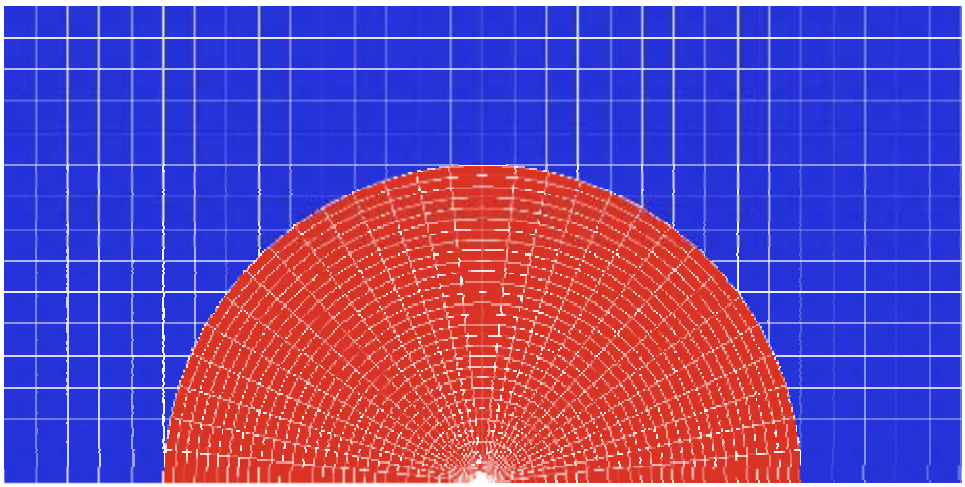

Fig. 7 - The computational grid at the initial time

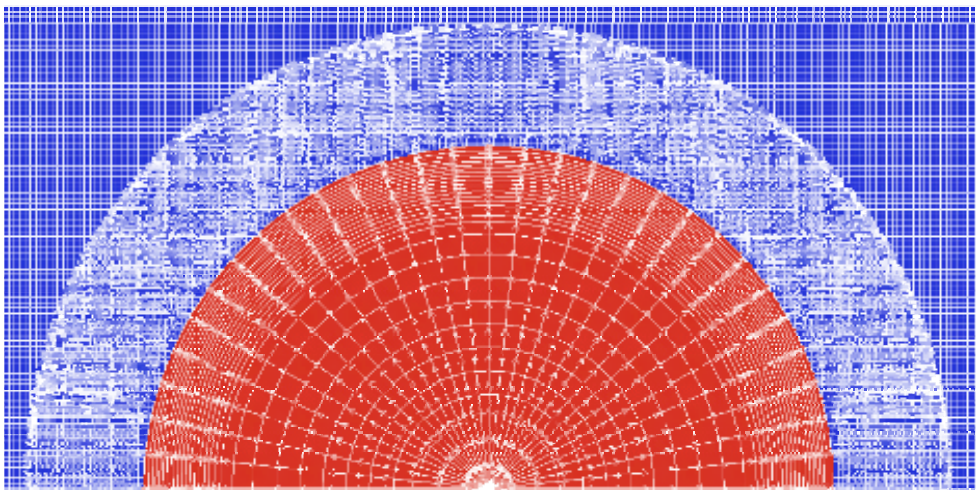

Fig. 8 – The computational grid at time 1.5

Presently the algorithms for communications between moving and unmoving fragments of the combined meshes are being improved further.

Originally, in the technique RAMZES-KP the radial direction was purely Lagrangian, which led to severe distortions of the computational grid in problems involving heavy distortions of physical region interfaces. So, to reduce crashes of the technique, algorithms for local refusal of the Lagrangian feature in the radial family of the computational grid have been developed and implemented into software. These algorithms maintain the problem grid in a state acceptable for the computation by meeting the grid quality criteria. Besides, if the crash does occur, then the timestep is recomputed with making the timestep shorter.

Using these algorithms has made the code significantly more crash-free/hand-off (in some cases up to one-run computation), with the accuracy being not deteriorated. This crash-free/hand-off technology has made 2D convergence computations of some problems feasible.

Conclusion

The paper has discussed the technique RAMZES-KP designed for computing multi-component heat-conducting flows in Eulerian-Lagrangian coordinates on parallel distributed-memory computer systems. Also, the modern directions of further development of the technique are presented.

References

1. Sofronov I.D., Voronin B.L., Bykov A.N., Skrypnik S.I. et al. "A technique and program complexes RAMZES, RAMZES-KP". Voprosy Atomnoy Nauki i Tekhniki. Ser. Matematicheskoye Modelirovaniye Fizicheskikh Protsessov 1999. Issue 4.
2. Voronin B.L., Bykov A.N., Veselov R.A. "Implementation of Yangs metod of interface tracking in three dimensions on orthogonal grids" Voprosy Atomnoy Nauki i Tekhniki. Ser. Matematicheskoye Modelirovaniye Fizicheskikh Protsessov 1999. Issue 3.
3. Noh V.F. SEL: Simultaneous Eulerian-Lagrangian method for computing 2D nonstationary problems. ACADEMIC PRESS, NEW YORK AND LONDON, 1964
4. Sedov L.I. Continuum mechanics. Moscow, Nauka Publishers. 1973, V. I, II, Edition 2.