Bull. Nov. Comp. Center, Num. Anal., 9 (2000), 29–32

Algorithms for the construction of quasiregular hierarchical grids*

G.A. Borisov, V.P. Il'in, V.M. Sveshnikov

1. Introduction. The problem of the construction of "good" grids for the numerical solution of multidimensional boundary value problems (BVPs) in complicated computational domains has a big history and extensive special literature, see [1–5], for example.

There are two main conventional requirements for the discretization of BVPs. The first one consists in the approximation quality which lies in the adaptivity to boundary peculiarities (vertices, edges, surfaces) and to differential properties of a solution, with possibility of local refinements. The second requirement means a simple grid topology and data structure to provide an efficient finite element or a finite volume approaches. We also take into account that modern fast algorithms of the numerical solution are based on the domain decomposition and multigrid principles which require much more complicated grid objects.

Thus, our objective is to satisfy the above mentioned conditions by constructing quasiregular, hierarchical, locally modified grids, for sophisticated two-dimensional boundary value problems.

The latter means that a computational domain (CD) consists of various computational subdomains (CSDs) with different functional or "physical" properties (differential equations to be solved with a description of coefficients and functions), and their boundaries can be piece-wise smooth, multiconnected and consisting of different segments of computational boundaries segments (CBSs).

A local modification, see [5], means that the grid nodes near boundary vertices and edges are shifted to the boundary with a possibility of saving the topology of a grid.

Quasiregularity means that a discretized computational domain, or grid domain, can consist of grid subdomains (GSDs), or subgrids, both regular and irregular. A regular grid, for example rectangular or triangular, is defined by its coordinate lines, and by simple uniform interconnections between the neighbouring nodes. Each GSD can include several embedded

*Supported by the NWO-NL-RF under Grant 047.008.007, the Russian Foundation for Basic Research under Grants 99-01-00579, 99-07-90422, and Grant of the Program "Universities of Russia" 99-10-59.

subgrids with refined meshsteps. We suppose three levels of grid refinement embeddings. This approach also includes the classical multigrid principle.

Formally, the problem of construction of a universal mesh consists in forming the grid data structure (GDS) which defines uniquely all the topological and geometric specifications of elementary grid objects, i.e., nodes, edges and finite volumes with necessary references to macro-objects from input data (CSDs, CBSs), which are the results of user graphic interface preprocessing. Thus, in what follows we present definitions and data structures for the macro- and the micro-level grid objects, algorithms a local of grid modification and data transformations which provide the full necessary information for the implementation of approximation of the BVP by finite volume or finite element methods.

2. Macrolevel data structure. Let $\Omega = \bigcup_k \Omega_k$ be an open computational domain of the BVP and $\overline{\Omega} = \Omega \cup \Gamma$ be its closure with the external boundary Γ. Here Ω_k, $k = 1, \ldots, N_{\mathrm{CSD}}$, are computational subdomains with different functional, or physical, properties (we also formally define Ω_0 as an exterior subdomain). The boundary $\Gamma = \bigcup_l \Gamma_{l,0}$, $l = 1, \ldots, N_{\mathrm{BS}}$, consists of different computational boundary segments $\Gamma_{l,0}$ which are specified by their own beginning and end points P_e^b, P_l^e, geometry (equations), and boundary conditions. The joint (internal) boundary segments $\Gamma_{l,l'}$ between the subdomains Ω_l and $\Omega_{l'}$, together with the external segments, form the boundary $\Gamma_l = \bigcup_{l'} \Gamma_{l,l'}$ of the subdomain Ω_l.

In addition to computational geometric objects we define the grid domain Ω^h, the boundary Γ^h, the subdomains $\Omega^h_{k'}$, $k' = 1, \ldots, N_{\mathrm{GSD}}$, $\Omega^h = \bigcup_{k'} \Omega^h_{k'}$ and the grid boundary segments $\Gamma^h_{m,m'}$ (between the BGSDs with numbers m and m') which provide the boundary $\Gamma^h_m = \bigcup_{m'} \Gamma^h_{m,m'}$ of Ω^h_m.

Computational subdomains and boundaries can coincide with the corresponding grid objects, but in general they can be different. The main conditions for their definition are: $\overline{\Omega} \subset \overline{\Omega}^h$, and Γ is approximated by a certain set $\Gamma^h_{m,m'}$.

For the sake of the efficient construction of algorithms, we also define computational-grid objects (CGSDs and CGBSs) which are the intersections of the respective computational and grid subdomains or boundary segments.

Each grid subdomain is formally defined as triangular, quadrangular (rectangular, in particular) or polygonal and its sides are described as a set of CGBSs.

Additional data include information on discretization of each side of GSD and the rule of definition of the nodes inside the GSD. For example, the 1D partition of sides can be piecewise analytical, i.e., uniform or with geometric or arithmetical increments.

3. Microlevel data structure. The elementary 2D grid objects are the nodes P_k^h, the edges E_l^h, and the non-intersecting volumes V_m^h (elements).

The grid domain is defined as $\Omega^h = \bigcup_m V_m^h$, and the grid boundary is a union of some set of the grid edges E_l^h.

Each node is defined by its coordinates, and affiliation to some boundary segment or a computational subdomain. The edges and the finite volumes are specified by their vertices as well as by relations with geometric macroobjects. Microlevel data structure also includes switch-back references on different types of elementary objects to provide all necessary functional information for the approximation procedures.

4. Local grid modification. One of the ways of constructing a regular grid adapted to a complicated computational domain is the following. A regular grid domain (GD, grid rectangles, for example) is superimposed in the CSD, and later the nearboundary nodes are shifted to the closest points of a boundary. There are two main rules for the resulting grid: each boundary vertex must be a grid node and all intersections of the boundary with grid edges must be nodes as well.

The grid modification of the algorithm can be formulated as follows. If some boundary vertex is disposed into the Dirichlet–Voronoi cell (the DVC), corresponding to a certain original (nonmodified) node, this node is shifted to the vertex. Also, if a smooth part of the boundary crosses the DVC, its node is shifted to the closest point of the respective part of the boundary. The results of the modification are the new coordinates of the shifted nodes and their encoding, i.e., equipping of the numbers for the respective boundary vertices or segments. In the modified and the original grid subdomains, the numbers of nodes and their links are the same, i.e., we have two topologically equivalent grids.

The algorithm is successively realized for each subdomain by loops on the corresponding internal grid-computational segments, and is in discretization of these segments and on coding (coloring) of the respective boundary grid nodes.

5. Algorithms of grid data structure transformations. Formally, any grid is defined by its data structure. On the microlevel, we consider two dual DSs, i.e., node and element (volume) oriented.

The DS node includes the numbering and the classification (encoding, or coloring) of grid points which are divided in to the following types: macronodes, i.e., coinciding with the boundary vertices, boundary and internal nodes of the first level (a coarse grid) and internal nodes of the second and the third levels (intermediate and fine grids).

We make use of two types of node numbering (ordering): regular and unique. The first one means the numbering of all the nodes in each grid

subdomain successively, i.e., for the first GSD, for the second GSD, etc.

In the GSD ordering, the high level grid subdomains succeed the parents subdomains of lower levels. Thus, in the regular ordering, one node with affiliation to different GSD, has several numbers. Another result of this approach consists in the formal definition of such a grid object as GSDs at different levels.

In the unique ordering, each node has only one number, and the corresponding list (the integer array) contains all the information on the node classification (coloring) and on the references to the regular numbering.

The subdomain affiliation of the node is defined by the algorithm of the point localization and by the geometric analysis of intersections of the coordinate lines with the boundaries of subdomains.

The element oriented data structure is constructed on the basis of the node DS, also by loops on grid subdomains and of the geometric analysis corresponding to computational boundaries and subdomains. Each element has a unique number only and contains all the geometric and functional information on its vertices, edges and volumes, necessary for approximation of the BVP and for formation of the algebraic data structure for the subsequent numerical solution to the problem.

References

[1] Liseikin V.D. A survey of methods for construction of structure grids // J. Comp. Math. and Math. Phys. – 1996. – Vol. 36, № 1. – P. 3–41.

[2] Kozlovsky G. Solving partial differential equations using recursive grids // Appl. Num. Math. – 1994. – № 14. – P. 165–181.

[3] Serezhnikova T.I., Sidorov A.F., Ushakova O.V. On one method of construction of optimal curvilinear grids and its applications // Sov. J. Num. Anal. Math. Mod. – 1989. – Vol. 4, № 2. – P. 137–155.

[4] Gurieva Y.L., Il'in V.P., Larin M.R. Inner data structures of the 2D boundary value problems. – Novosibirsk, 1997. – (Preprint / RAN. Sib. Branch. Comp. Cent.; 1090) (in Russian).

[5] Matsokin A.M. Automation of triangulation of subdomains with a smooth boundary for solution of elliptic equations.– Novosibirsk, 1975. – (Preprint / RAN. Sib. Branch. Comp. Cent.; 15) (in Russian).