Bull. Nov. Comp. Center, Num. Anal., 8 (1998), 71–79

The program of Delaunay triangulation construction for the domain with the piecewise smooth boundary*

I.A. Sander

1. Introduction

In this paper we describe the program of triangular mesh construction with Delaunay properties for the domains, which boundaries consist of straight lines and arcs of circles and ellipses. There was used the geometric preprocessor TTNS [1] in order to form the discret similarity for the calculated domain and to construct its finite-difference approximation. This preprocessor. produces the representation of mesh domain not as some knots variety, but as a variety of mesh lines segments, completely belonging to the domain, with the final points on the boundary. So, it is possible to reach a linear dependence of computational time and memory required on the mesh-size parameter because of the way of final data representation: it is $O(h^{-1})$ value.

The program suggested earlier for the triangulation of domains with the piecewise smooth boundary [1] started from the attempt to process the domain in general with the help of the local modification algorithm [2, 3], suited for the domain with a smooth boundary. So, it was necessary to provide in the input data such relations between the boundary of calculated domain and the rectangular mesh, that the localization of all corner points of boundary at the knots of mesh would be possible. There was also an essential restriction to the mutual location of boundary curves, because there was impossible to process the neighborhood of the corner point, formed as an intersection of numerable boundary curves. Their maximum number, even in "favorable" situation, was restricted with eight curves, but in general case with two only.

This program is based on the algorithm suggested by S.A. Sander. According to this algorithm the initial domain must be automatically decomposed to the basic subdomain and its supplements in the process of domain triangulation. Besides the triangle mesh construction in the basic domain deals with transformation of an initially defined rectangular mesh into the

*Supported by the Russian Foundation for Basic Research under Grant 98-01-00709.

quadrangle one with the following subdivision of its cells to triangles. The classical algorithm of the Delaunay triangulation construction was realized for the supplements [4].

A special overlap is exracting to make consistent the triangulation in general. There are the cells of quadrangle mesh, relating to the basic subdomain and adjoining to the supplement in the overlap. So, the initial variety of points of triangulation involvs not only outsiding for the basic domain knots, but also all knots in the overlap.

2. *A* operator of the local displacement of nearboundary knots

An initial rectangle mesh must be converted into the quadrangle one so, that only at the tops of its cells would be located both the cross-points of boundary curves (corner points) and the points of intersection of boundary curves with the cell sides.

Let us designate the totality of the modified mesh knots as

$$M = \{m = (i_m, j_m)\},$$

where i_m, j_m are the numbers of mesh lines crossing at m knot.

Let the boundary points (the points in set T_Γ) be the points of intersection of boundary curves with lines of initial rectangle mesh, as well as all boundary corner points, placed out of mesh lines too. Then M_Γ would be defined as the subset M, insists of candidates for displacement to one of the points in T_Γ:

$$M_\Gamma = \bigcup_{t \in T_\Gamma} M_t, \qquad M_t = \{m \in M : \ |x_t - i_m| \le 0.5, \ |y_t - j_m| \le 0.5\},$$

where x_t, y_t are the relative coordinates of the point t. For each knot in M_Γ we define the set T_m, $m \in M_\Gamma$ - the set of points T_Γ, where m knot could be displaced: $T_m = \{t \in T_\Gamma : \ M_t = m\}$.

Let us call the set T_m as a set of attractive points for m knot. If there is a corner point as one of the elements in the set T_m, its power could be reduced by means of elimination of points, located on forming this corner point bounary curves, but not this corner point itself.

The "most attractive" point t_m in the set T_m will be the corner point, if it is, otherwise

$$t_m = \arg\min_{t \in T_m} |x_t - i_m| + |y_t - j_m|. \tag{1}$$

If there are several points in T_m, complying with condition (1) and locating on the same boundary curve, each of them may be used as the most attractive point. In addition, when $|x_t - i_m| = 0.5$ or $|y_t - j_m| = 0.5$, the point

t may be used as the most attractive point only if $i_m < x_t$ and $j_m < y_t$ respectively.

The inquiries to displace some knot m to the various boundary curves will be contradictory. If there are no contradictory inquiries in all sets T_m, $m \in M_\Gamma$, the displacements to the most attractive points $A : m \to t_m$ will modify an initial rectangle mesh so, that the sides of its cells approximate the boundary with the second order and the span deviations from the congruent boundary arcs do not exceed $O(h^2)$. There may be a part of boundary as a diagonal in some modified cell. If the length of this diagonal is not minimal for this cell, the attempt of triangulation of a quadrangle mesh would be accompanied with the contradictory inquiries to this cell. So, the A operator of local displacement of nearboundary knots is not finished the modification of an initial rectangle mesh.

3. *B* operator of the local displacement

In order to construct a Delaunay triangulation, based on a modified mesh M, we need in the following deformation of a quadrangle mesh in the way of some knots displacing. The violation of the Delaunay condition may be in the cells, where the approximated boundary diagonal is not minimal. Knots at the top of such cells may be subjected to B operator effect. Thus we can improve the mesh quality not because of the knots displacement to the boundary, but because of displacement of the knots that probably are not next to the boundary, inside the mesh domain. Both the direction and the size of B–displacement of a knot depend on a mutual location of knots inside the quadrangle cell, while the A–displacement of a knot was determined by its location in respect to the domains boundary.

We can exhaust the set of cells, generating B-displacements, consists of four types of the cells to within theirs turning (Figures 1–4). Knots c and d are A-displaced in each case.

In the first case the equally directed vectors of displacement of the knots c and d produce B-displacement of the knot n, equal to the displacement of c in size and direction (see Figure 1):

$$Ac = c + \alpha e_c, \quad Ad = d + \beta e_d, \quad Bn = n + \alpha e_c.$$

There are two outsiding A-displaced knots in the cell in the second case, generating B-displacements of the couple of knots, where the displacements of c and d are the prototypes for n and m respectively (see Figure 2):

$$Ac = c + \alpha e_c, \quad Ad = d - \beta e_d, \quad Bn = n + \alpha e_c, \quad Bk = k - \beta e_d.$$

The belonging to the third type of the cell is defined by the fact that there was a right angle between the vectors of c and d displacements, while one of these knots was moved inside the cell and the other one – outside.

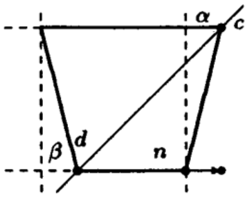

Figure 1

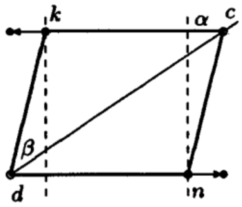

Figure 2

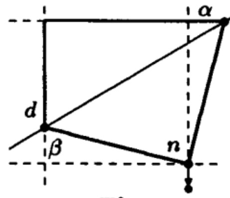

Figure 3

B-displacement of n must be opposite to A-displacement of the knot d and equal to the displacement of c in size (see Figure 3):

$$Ac = c + \alpha e_c, \quad Ad = d - \beta e_d, \quad Bn = n + \alpha e_d.$$

There is a singularity for the fourth type of the cell, where the outsiding A–displacements of both diagonal knots are perpendicular to each other.

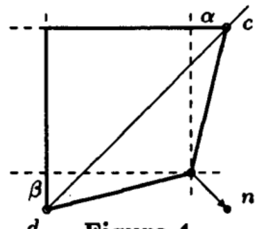

Figure 4

It is necessary to displace the knot n not along some of mesh lines, but along the vector, summing the initial A-displacement of the knots c and d (see Figure 4):

$$Ac = c + \alpha e_c, \quad Ad = d - \beta e_d,$$
$$Bn = n + \alpha e_c - \beta e_d.$$

Let us assume that the set of new founded B-points based on a some fictitious boundary curve Γ_0 and joins it with the set T_Γ. Let there be some set T_m with the nonunique point $t_b \in \Gamma_0$, corresponding to the inquiry to B-displacement of the knot m. If there is also $t_a \in T_m$, displaced in the same direction, but farther distant from the knot m, then it will be $t_m = t_a$ as a most attractive point, while the power of T_m will be reduced because of t_b eliminating.

4. The basic subdomain triangulation

The following data processing is connected with the elimination of non-contradictory inquiries in order to have not more than one point for one boundary curve in T_m, $m \in M_\Gamma$. The point of a most attraction among all A-displacements is the nearest one to the knot m for each given curve. We can reduce the power of the set T_m because of B-displacemets elimination too. If there are several B-points located on a ray, started at the knot m, it will be sufficient to fix the most remoted among them.

This algorithm is based on the attempt of processing of maximal possible part of domain with the help of local modification algorithm. So we need to select the knots of modifying the mesh being in contradiction to this algorithm. First, these are knots with contradictory inquiries to displace, i.e., the power of an appropriate set T_m is more than unit. Second, these are knots having a corner point located out of mesh lines as a point of attraction, even if such point is a single one.

The zone of contradictions is automatically generated for every such knot. It is a polygon, consisting of quadrangle cells with this knot among their tops. These polygons must be eliminated from the basic subdomain in order to form some supplementary subdomain. We need to check the Delaunay condition for the triangles in the basic subdomain next to the outline of supplementary subdomain. Its violation is the reason to carry an appropriate cell from the basic subdomain to a supplementary one.

Having all displacements in the basic subdomain done, an initial rectangle mesh is modified in a such a way, that edges and diagonals of their cells approximated all boundaries with the second order. To complete the construction of the mesh triangulation we need to divide all quadrangle cells into two triangles with one of their diagonals. Let us call the cell with although one of its top displaced to the boundary as modified. The unmodified cell should be divided into two triangles with its left down corner diagonal. If there are two tops in a modified cell, placing diagonally and belonging the same boundary curve, the choice of diagonal is evident. It is the segment between these displacing tops. If such a couple is absent, we need to choose the minimal diagonal in a cell.

Let us see in detail the structure of output data for the basic subdomain triangulation. Its singularity is in using $O(h^{-1})$ numbers to describe the mesh with $O(h^{-2})$ triangle elements. In order to consider the specific character of a triangle mesh constructed on a base of a locally modified mesh we need to introduce the term of stripe as a unit of a triangle mesh description. The stripe here is the part of calculated domain, situated between two adjacent x-lines of a quadrangle mesh and allocated in one of its subdomains entirely. These x-lines are the broken line in fact as a result of some knots displacing. There may be three fragments in a stripe. The first one is in the beginning of a stripe and consists of the modified cells entirely. The second fragment consists of the unmodified cells only, if they are in the middle of this stripe. The third fragment at the end of the stripe consists of the modified cells, like the first one. If there are no unmodified cells in the stripe, it will be the one fragment stripe. The amount of triangles both in the stripe and its fragments is unlimited.

We need in one date for each x-line of a mesh, meaning the amount of the stripes, adjoining this line from the right. The output data confined oneself to such parameters:

- the number of the subdomain corresponding;
- the number of the y-line of a mesh, limiting this stripe from below;
- the number of cell in all fragments of a stripe.

There is no additional information about the unmodified cells. As for the modified cells, we need such an information, equal to their amount in size. According to the way of the cell intersection and the amount of triangles in it we can extract 6 types of the modified cells. If the triangles in one cell are from various subdomains, such a cell would be in two stripes description.

5. The supplementary subdomains triangulation

Each supplementary subdomain should be increased at the expense of adjoining cells from the basic subdomain in order to reach the compatible triangulation in a whole domain. So, there is an overlap with the cells both in basic and supplementary subdomain. The special set of triangulation points consists of all displacement inquiries for all knots in a supplementary subdomain and all tops of triangles in the overlap of this subdomain.

We can construct the triangulation in each supplementary subdomain with the help of recursive algorithm mentioned by Mock [4]. It uses the method of artificial limitation of a domain, and some artificial points are required to construct a convex hull with all points from the special set in it. Setting the Delaunay decomposition for these artificial points done we can add real points from the special set one by one. All elements of trianglulation with artificial points must be rejected afterwards.

Having the Delaunay triangulation done, we need to check the approximation of all boundary curves with the sides of triangles. We can correct the set of points if necessary. If there is such a boundary segment, that couldn't be approximated with the help of initial set of points, all points from this segment must be eleminated. The triangles having these points as their tops must be eleminated too. The polygon formed by these triangles must be divided into new triangles. The number of fragmentation for the unapproximated segment of boundary should be increased and the appropriate points adjoned to the set. The choice of new points has been realized so, that none of them could fail the triangles having approximating boundary sides.

There is a need to identify all the triangles in the ovelaps to obtain a compatible triangulation for the domain on the whole. We can hope for it, because there is the Delaunay triangulation in a basic subdomain and the points from the special sets couldn't disturb it. Nevertheless, there may be some conflicts for some triangles forming the Delaunay polygons. We can obtain a few equivalent sepatation in this case and as a dominating one will be those from the basic subdomain.

Some tests were performed with the sequential condensation of mesh for the domains of the various configuration. It is possible to get the triangle mesh with 400000 points in 0.1 s of CPU time (IBM PC/AT 486SX) for the domains of relatively simple configuration (Figures 5–7).

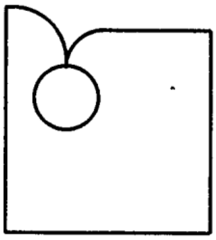

Figure 5

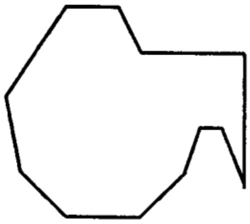

Figure 6

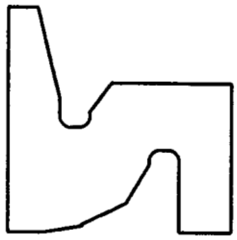

Figure 7

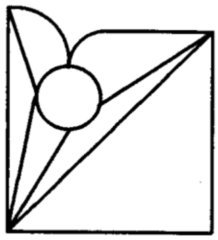

Figure 8

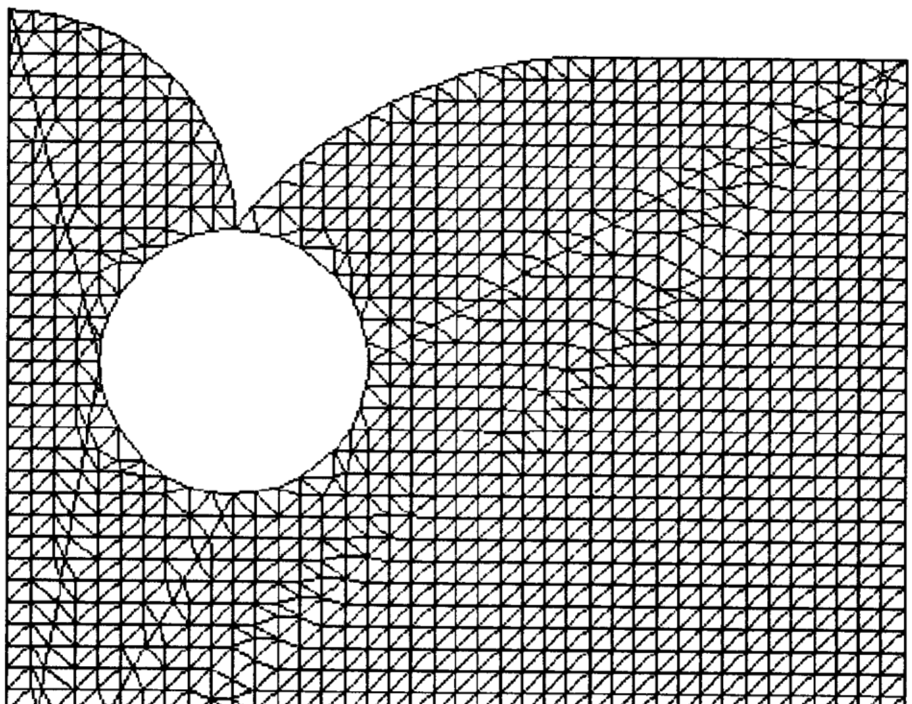

Figure 9

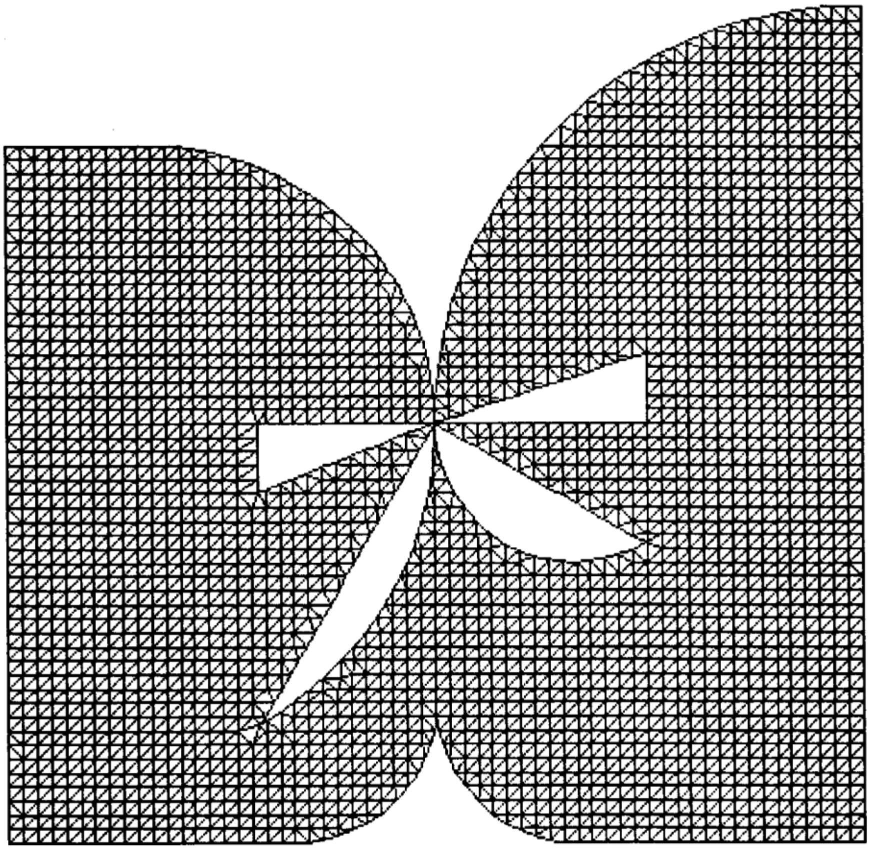

Figure 10

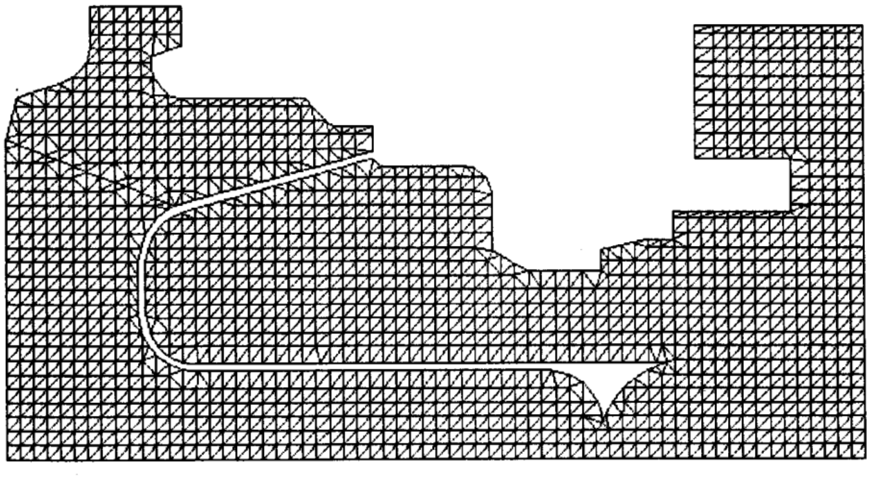

Figure 11

There are some factors that can cause the complication of configuration of domain calculated. The first one is in the presence of the intrinsic boundaries, increasing the amount of stripes in the basic subdomain. The next one is the existence of the corner points, formed as a point of intersection of the numerable boundary curves. The zones of contradictions are generated for such points, and the time their triangulation is the $O(k^2)$ value, where k is the amount of points in the zone of contradictions. We need to note that the amount of such points does not increased in the process of mesh condensation on the whole. Moreover the time of mesh generation could be reduced as a result, because of an increase of the supplementory subdomains inside the same set of points.

It needs 0.44 s to get the triangle mesh with 50000 points for the domain with 5 intrinsic boundaries and 7 subdomains in it (Figures 8–9). The condensation of triangle mesh (Figure 10) up to the 67000 points required 0.30 s, while the mesh with 130000 points (Figure 11) was generated in 1.5 s. There is a linear dependence of the computational time and memory required on the mesh-size parameter for the basic subdomain in the process of mesh condensation.

References

[1] I.A. Sander, *The triangle mesh generation for the domains with the piecewise smooth boundary*, Comp. Cent. Sib. Branch, USSR, Acad. Sci., Preprint No. 995, Novosibirsk, 1993 (in Russian).

[2] A.M. Matsokin, *The automatic triangulation of the domains with the smooth boundary for solving the elliptic equations*, Comp. Cent. Sib. Branch, USSR, Acad. Sci., Preprint No. 15, Novosibirsk, 1975 (in Russian).

[3] V.V. Shajdurov, *The Multigrid Methods of Finite Elements*, Nauka, Moscow, 1989.

[4] M.C. Mock, *The tetrahedral elements and the Scharfetter–Gummel method*, The numerical analysis of semiconductor devices and integrated circuits, Radio i svjaz, Moscow, 1989, 90–98 (in Russian).