Automatic Discretization of a Three-Dimensional Domain into Voronoi Polyhedron Elements*

Mutsuo KOBAYASHI**, Hiroshi MAEKAWA**
and Yasuhira KONDOU**

When a partial differential equation is solved numerically, it is conventional to discretize the calculating domain into many elements by elaborate handwork. For this discretization we devised a new method with which the three-dimensional domain can be automatically divided into Voronoi polyhedron elements enclosing arbitrary nuclear points. The present report describes this method in detail giving an example. When the Voronoi polyhedron elements are used, various types of differential equations can be easily transformed into simultaneous algebraic equations for each of the nuclear points, and parallel processing can be achieved more quickly and accurately. Moreover, the Voronoi polyhedron element table has possibilities for future development: it can be used not only as a direct analytical method of solving a partial differential equation, but also, after being transformed, as an ordinary tetrahedron element table.

Key Words: Grid Generation, Finite-Element Method, Computer-Aided Design, Finite Volume Method, Numerical Analysis, Voronoi Polyhedron, Automatic Discretization, Parallel Processing

1. Introduction

To solve a partial differential equation numerically, it is necessary to transform the equation into discrete simultaneous algebraic equations by dividing the calculating domain into many subdomains (elements). Since this discretization is usually performed by elaborate handwork, presently, most of the processing time is consumed by these transformations. As reviewed by Oda and Yamazaki⁽¹⁾, some progress has been made in the automatic discretization for the two-dimensional domain by means of Green and Sibson's⁽²⁾ method using the Voronoi polygon and by other methods⁽³⁾. However, for the three-dimensional domain, no concrete method for the automatic discretization has been proposed, except for Bowyer's⁽⁴⁾ and Watson's⁽⁵⁾ methods using the Voronoi polyhedron and the Delaunay tetrahedron. For the case where the element points are randomly located in the two-dimensional domain, the authors⁽⁶⁾ have already devel-

oped a new automatic discretization method using the Voronoi polygon, and have provided an example showing the simple and accurate transformation of the Laplace equations. Recently, Taniguchi et al.⁽⁷⁾ also used such an element shape in the analysis of a flow field. In the present paper, the new automatic discretization method, developed by the authors⁽⁶⁾, is extended to the case of the three-dimensional domain, and a concrete example is described in detail.

2. Voronoi Polyhedron

According to the Science Dictionary published by Maruzen⁽⁸⁾, the Voronoi polyhedron is a concept that indicates the distinctive features of particle arrangement by means of the discretizing of the space, and is the shape obtained by joining each point in the space to its nearest particle. In order to provide a concrete demonstration, we give some arbitrary nuclear points (element points), and divide (discretize) the space by the perpendicular bisectors of neighboring points. The characteristics of this polyhedron are as follows:

(1) A boundary plane of the polyhedron is a perpendicular bisector of the line segment which joins the element point to a neighboring element point.

* Received 27th November, 1991. Paper No. 90-1120 A

** Faculty of Engineering, Niigata University, Igarashi 2-8050, Niigata 950-21, Japan

Table 1 Element table of the Voronoi polyhedron

Element Name I=7		Element Coordinates X(7),Y(7),Z(7)		Number of Faces FC(7)=10	
Forefront Face Address FI(7)=84 + J		Forefront Address EI(7)=645 + K			
Face Address	J	84	85	86	-
Face Name	F(J)	24	11	18	-
Number of Sides	EC(J)	4	5	4	-
Side Address	K	645, 646, 647, 648	649, 650, 651, 652, 653	654, 655, 656, 657	- , -
Side Name	E(K)	- , - , 11, 33	24, - , - , 18, 33	- , 33, 11, -	- , -

Fig. 1 Flow chart of the automatic discretization

(2) A side of the polyhedron is the line segment which is equidistant from the element point and two neighboring element points.

(3) A vertex of the polyhedron is the point which is equidistant from the element point and three neighboring element points.

(4) When a new point is arbitrarily given in the space divided into the Voronoi polyhedra, the closest element point to this point is that of the Voronoi polyhedron which contains this point.

(5) When the element points are located arbitrarily, the obtained polyhedra are convex, and their shapes vary according to the arrangement of the element points.

3. Automatic Discretization Method

Contrary to the methods devised by Bowyer⁽⁴⁾ and Watson⁽⁵⁾, the method of the present authors uses tables (element tables) which directly provide the Voronoi polyhedra enclosing the given element points. By referring to the flow chart of the discretization procedure shown in Fig. 1, it is found that the domain is automatically discretized by adding a new element point arbitrarily, and by assigning its coordinates. A

Fig. 2 Voronoi polyhedron element

detailed description of this flow is given in the following concrete example.

3.1 Element table

One of the essential features of this discretization program is the use of the element table. The polyhedron element named $I=7$ according to its order of generation is shown in Fig. 2. The corresponding element table is Table 1. Each individual Voronoi polyhedron is expressed by the coordinates of the element point, number of faces, face names, number of sides (number of vertexes) and side names, all of which are presented in the element table. The face name is defined as the name of the neighboring element which is on the side opposite to the face. One of the sides constituting the face is equidistant from three element points: the element point under consideration, the element point of the face name, and another element point; thus, the name of the last element point is used as the name of this side. Accordingly, the side name indicates the name of a face which is linked with the face by the side. The face names and the side names are memorized in the array variables, and have the respective subscripts of the face address and side address which are numbers starting from element 1 (element with the element name of 1). The side addresses are numbered clockwise from the viewpoint of the element point.

3.2 Setup of the initial data

We obtain the Cartesian coordinates as shown in Fig. 3, and define the dividing domain as the inside of a cube with its center at the origin. As shown in Fig. 3, the six faces (frame faces) of the cube are distinguished from the other faces by their negative names (-1 to -6). To begin with, we generate element point 1 at an arbitrary position within the dividing

Table 2 Initial element table

Element Name I=1		Element Coordinates X(1),Y(1),Z(1)				Number of Faces FC(1)=6	
Forefront Address FI(1)=1 → J			Forefront Address EI(1)=1 → K				
Face Address	J	1	2	3	4	5	6
Face Name	F(J)	-1	-2	-3	-4	-5	-6
Number of Sides	EC(J)	4	4	4	4	4	4
Side Address	K	1, 2, 3, 4	5, 6, 7, 8	9,10,11,12	13,14,15,16	17,18,19,20	21,22,23,24
Side Name	E(K)	-3,-5,-4,-6	-3,-6,-4,-5	-1,-6,-2,-5	-1,-5,-2,-6	-1,-3,-2,-4	-1,-4,-2,-3

Fig. 3 Frame faces and Cartesian coordinates

Fig. 4 Addition of a new element point

Fig. 5 Processing element (element 2)

domain. Then, element 1 is the cube itself and the corresponding element table (initial element table) becomes Table 2. As a concrete example shown by Fig. 4, we assume the situation in which element point 3 (additional element point) is added to the domain when the number of elements is two.

3.3 Search for the mother element

An element which has an additional element point inside itself is called a mother element. (This is the element 2 in the present example.) Characteristic (4) of the Voronoi polyhedron can be used to search for the mother element. The comparison of the distances between the additional element point and every other element point is not necessary; the mother element can be detected efficiently by the following method. At

first, we obtain a temporary mother element arbitrarily. If there is a neighboring element closer to the additional element point than the temporary mother element point, then we replace the temporary mother element with it. When this is not the case, the temporary mother element is the true mother element. This mother element is the first of the processing elements described in the following section.

3.4 Determination of the vertex attribute of the processing element

The processing element (PE) is the element which neighbors on (is in contact with) the additional element (AE). It is discretized into the shared element (SE, fragment of AE) and the renewed element (RE, remainder of PE). The processing element is cut with the perpendicular bisector of the line segment which joins the element point to the additional element point. First, the vertex attribute of the processing element is determined by comparing the distances from the vertex to the processing element point and to the additional element point. By this process, the coordinates of the vertex are obtained using characteristic (3) of the Voronoi polyhedron, whereby the vertex is equidistant from four points whose names are the processing element name, the additional element name and two names of sides containing the vertex. When the side name is negative, the coordinates of the vertex in question are obtained, noting that it lies on the frame (plane or side) and is equidistant from (three or two) other element points.

As an example to illustrate the determination of the vertex attribute, we consider the first processing element (element 2) shown in Fig. 5 and the element table shown in the upper part of Table 3. Let the initial address ($L=1$) of the array variable $G(L)$ of the vertex attribute coincide with the forefront side address $EI(I)$ in the processing element table. Furthermore, let the addresses of $G(L)$ correspond with the side addresses of the processing element in order. The vertex names of the surface polygon constructed by the sides can be substituted for the side names. However, in Fig. 5, the vertex names are expressed by those of the Voronoi polyhedron: A, B, C, D, E, F, G and H. The vertex attribute $G(L)$ is determined as follows.

(1) If the vertex is in the side of the shared

Table 3 Processing element table (element 2) and determination of attribute

Element Name I=2		Element Coordinates X(2),Y(2),Z(2)			Number of Faces FC(2)=6		
Forefront Address FI(2)=14 + J		Forefront Address EI(2)=43 + K					
Face Address	J	14	15	16	17	18	19
Face Name	F(J)	1	-1	-2	-4	-5	-6
Number of Sides	EC(J)	4	4	4	4	4	4
Side Address	K	43,44,45,46	47,48,49,50	51,52,53,54	55,56,57,58	59,60,61,62	63,64,65,66
Side Name	E(K)	-6,-2,-5,-1	1,-5,-4,-6	1,-6,-4,-5	-1,-5,-2,-6	1,-2,-4,-1	1,-1,-4,-2
Vertex Name	(Fig.5)	B, C, D, A	B, A, E, F	C, G, H, D	F, E, H, G	A, D, H, E	C, B, F, G
Vertex Address	L	1, 2, 3, 4	5, 6, 7, 8	9,10,11,12	13,14,15,16	17,18,19,20	21,22,23,24
Vertex Attribute	G(L)	1,-1,-1, 1	1, 1, 1, 1	-1,-1,-1,-1	1, 1,-1,-1	1,-1,-1, 1	-1, 1, 1,-1
Reference Address	M	1	2	3	4	5	6
Face Attribute	D(M)	0	1	-1	0	0	0

Fig. 6 Shared face fragment

Fig. 7 Divided face

Fig. 8 Renewed face fragment

element, we assign $G(L)$ the value of 1; (closed circles A, B, E and F).

(2) If the vertex is in the side of the renewed element, we assign $G(L)$ the value of -1; (open circles C, D, H and G).

(3) If the vertex lies on the boundary plane between the shared element and the renewed element, i.e., if it is equidistant from their element points, we assign $G(L)$ the value of 0.

Equidistance must be determined taking into account the rounding error in the calculation.

After all the vertex attributes of the processing element are determined, the face attribute is determined, based on the vertex attributes.

3.5 Determination of the face attribute of the processing element

The face attribute $D(M)$ indicates that the face belongs to either the shared element or the renewed element, or that the face is divided. $D(M)$ of the individual face of the processing element is determined according to the vertex attribute as follows.

(1) If every vertex attribute $G(L)$ of the face is equal to 1, the face attribute $D(M)$ is assigned to be 1, and the face itself becomes that of the shared element, leaving its figure as is.

(2) If every vertex attribute $G(L)$ of the face is not equal to 1, the face attribute $D(M)$ is assigned to be -1, and the face itself becomes that of the renewed element, leaving its figure as is.

(3) In cases other than the above-mentioned two cases, the face attribute $D(M)$ is assigned to be 0, and

the face is divided into two pieces. Accordingly, the face attributes of the example are determined as in Fig. 5 and Table 3.

3.6 Dividing process

The dividing process consists of three subprocesses: division of a face of the processing element into two fragments, distribution of the fragments and the undivided faces into the shared element or the renewed element and generation of the new section (boundary face) of those elements.

3.6.1 Dividing and distributing faces If the face attribute D is equal to zero, then the face (divided face shown by Fig. 7) is divided into the fragment of the shared element (shared face fragment shown by Fig. 6) and the fragment of the renewed element (renewed face fragment shown by Fig. 8) by the perpendicular bisector. Face names, side names and others of the former and latter fragments are accumulated into the shared element table and the renewed element table, respectively. In the example, the first divided face is found to be the first face in the memory address of the processing element. It is shown again in Fig. 7 and Table 5, and is divided as follows.

(1) We prepare the face column of 1 with the same name as the divided face name in the shared element table (Table 4). The first side of this face is named 2, which is the same as the processing element name.

(2) We prepare the face column of 1 with the same name as the divided face name in the renewed element table (Table 6). The first side of this face is

Table 4 Shared element table

Number of Faces CFN=1		
Face Address	L	1
Shared Element Name	CK(L)	2
Face Name	CF(L)	1
Number of Sides	CEC(L)	4
Side Address	M	1, 2, 3, 4
Side Name	CE(M)	2, -5, -1, -6

Table 5 Portion of the element table and attribute (divided face 1)

Face Address	J	14
Face Name	F(J)	1
Number of Sides	EC(J)	4
Side Address	K	43, 44, 45, 46
Side Name	E(K)	-6, -2, -5, -1
Dividing Point	:	:
Vertex Name (Fig.7)		B, C, D, A
Vertex Address	L	1, 2, 3, 4
Vertex Attribute G(L)		1, -1, -1, 1
Dividing Point	:	:

Table 6 Renewed element table

Number of Faces TFN=1		
Face Address	L	1
Face Name	TF(L)	3
Number of Sides	TEC(L)	4
Side Address	M	1, 2, 3, 4
Side Name	TE(M)	3, -6, -2, -5

Fig. 9 Order of dividing process

Table 7 Shared element table

Processing Element Name=2 → CK(L)			Number of Faces CFN=6					
Face Address	L	1	2	3	4	5	6	
Shared El. Name	CK(L)	2	2	2	2	2	2	
Face Name	CF(L)	1	-5	-4	-6	-1	2	
Number of Sides	CEC(L)	4	4	4	4	4	4	
Side Address	M	1, 2, 3, 4	5, 6, 7, 8	9,10,11,12	13,14,15,16	17,18,19,20	21,22,23,24	
Side Name	CE(M)	2,-5,-1,-6	2,-4,-1, 1	2,-6,-1,-5	2, 1,-1,-4	1,-5,-4,-6	-6,-4,-5, 1	

named 3, which is the same as the additional element name.

(3) We search for the side of the divided surface polygon which joins one vertex with $G(L) \neq 1$ to another vertex with $G(L+1) = 1$, and write the side name as the second side name in the face column of the shared element table. Furthermore, we write in order the side names of the divided face as the side names in the column until the side which joins one vertex with $G(L) = 1$ to another vertex with $G(L+1) \neq 1$ (see Table 4) is obtained.

(4) We search for the side of the divided surface polygon which joins one vertex with $G(L) = 1$ to another vertex with $G(L+1) \neq 1$. We write the side name as the second side name in the face column of the renewed element table, and write in order the side names of the divided face as the side names in the column until the side which joins one vertex with $G(L) \neq 1$ to another vertex with $G(L+1) = 1$ (see Table 6) is obtained.

(5) The face of the same name (-5 in the example) as the second side in the face column of the shared element is also the divided face. Accordingly, the face of the processing element is divided in the same way, using the above-mentioned process of steps

(1) to (4).

As shown in Fig. 9, the process of steps (1) to (4) is repeated until it returns to the first divided face. The next step, in order of these processes, is indicated by the second side name (the side name of the second number in the column) of the previous divided face name of the shared element table (see Table 7). In the present example, the shared element table accumulates data for the face addresses of 1 to 4 in Table 7, and the renewed element table accumulates data for the face addresses of 1 to 4 in Table 8.

3.6.2 Distribution of the undivided faces The undivided faces as shown in the left side of Fig. 5 are processed as follows. If the face attribute D is equal to 1, the data in the face column of the shared element table (face address 5 in Table 7) are copied from the face column of the processing element table (face address 15 in Table 3). If the face attribute D is equal to -1, the data in the face column of the renewed element table (face address 5 in Table 8) are copied from the face column of the processing element table (face address 16 in Table 3).

3.6.3 Generation of the boundary face As shown in Fig. 10, the new section is produced by cutting the processing element with the perpendicular

Table 8 Renewed element table

Element Name	I=2	Element Coordinates X(2),Y(2),Z(2)			Number of Faces TFN=6		
Face Address	L	1	2	3	4	5	6
Face Name	TF(L)	1	-5	-4	-6	-2	3
Number of Sides	TEC(L)	4	4	4	4	4	4
Side Address	M	1, 2, 3, 4	5, 6, 7, 8	9,10,11,12	13,14,15,16	17,18,19,20	21,22,23,24
Side Name	TE(M)	3,-6,-2,-5	3, 1,-2,-4	3,-5,-2,-6	3,-4,-2, 1	1,-6,-4,-5	1,-5,-4,-6

Table 9 Additional element table

Number of Faces	AFN=1
Face Address	L 1
Face Name	AF(L) 2
Number of Sides	AEC(L) 4
Side Address	M 1, 2, 3, 4
Side Name	AE(M) -6,-4,-5, 1

Fig. 10 Generation of boundary faces

Table 10 Frame table

Frame Face Name	-I	-1	-2	-3	-4	-5	-6
Number of Fragment	BC(I)	1	0	0	1	1	1
Face Address	J	1	-	-	1	1	1
Face Fragment Name	BK(I,J)	2	-	-	2	2	2
Number of Sides	BEC(I,J)	4	-	-	4	4	4
Top Side Address	BEI(I,J)	1	-	-	1	1	1
Side Address	M	1, 2, 3, 4	-	-	1, 2, 3, 4	1, 2, 3, 4	1, 2, 3, 4
Side Name	BE(I,M)	1,-5,-4,-6	-	-	2,-6,-1,-5	2,-4,-1, 1	2, 1,-1,-4

bisector, and is processed as follows. We prepare the face column 2 with the same name as the processing element, and write the face names processed in section 3.6.1 in this side name column (face address 6 in Table 7) in the order of processing. We prepare the face name column 3 with the same name as the additional element, and write the face names processed in section 3.6.1 in this side name column (face address 6 in Table 8) in the reverse order of processing.

3.7 Treatment of the shared faces

The faces accumulated in the shared element table as a result of the processing in section 3.6 are called the shared faces. They are treated as follows.

3.7.1 Treatment of the boundary face The shared face with the same name as the processing element, i.e., that generated in section 3.6.3 (face address 6 in Table 7), is copied in the additional element table (Table 9) which accumulates the data of the additional element.

3.7.2 Treatment of the frame face fragment The shared faces with negative names (frame face fragments) are accumulated in the frame table (FT). The additional element is that produced by gathering up the shared elements. Accordingly, when there are frame face fragments in the shared faces, the frame face fragments with the same name must be united (composed into one). This composing process is carried out after the accumulating process, and will be

Table 11 Processing table

Reference Address	JA	1, 2, 3
Processing Element Name	J(JA)	2, 1, 0
Finished Check	C(JA)	2, 2, 0

described in detail in section 3.10. In the case of the shared faces cut off from element 2, those with names of -1, -4, -5 and -6 are obtained. They are copied from the shared element table (face addresses 2 to 5 in Table 7) to the frame table (Table 10). At this time, the name of the original element before the division (shared element name $CK(L)$), i.e., the name of the element under processing, is written with the name of the frame face fragment ($BK(I,J)$) in the frame table.

3.7.3 Registration of the processing element The dividing process must be conducted for the elements with the same names as the other divided faces (i.e., faces with positive names except for the boundary face). Thus, the names of these elements are registered on the processing table ($J(JA)$ in Table 11). We search for an unprocessed element using this table, and continue the dividing process. The mother element name has been registered beforehand on the top $J(1)$ of the processing table, and the names of the elements which require the dividing process are registered on $J(JA)$ in succession. When $J(JA)$ are registered, the finish checks $C(JA)$ are given the processing

element names. When the dividing process of the element $J(JA)$ has been completed, $C(JA)$ is assigned a zero value. We examine $C(JA)$ in order of the address. When $C(JA)$ is not equal to zero, then $J(JA)$ is the element name for the next dividing process. If the face name of the divided elements newly obtained by the dividing process is equal to the previously registered $J(JA)$, it is not registered once more. The dividing process for the additional element point is continued until every registered name becomes zero.

3.8 Renewal of the processing element

The processing element changes its shape and becomes the renewed element by cutting off the shared element. Hence, the processing element table (Table 3) is replaced by the renewed element table (Table 8). If the size of the renewed element table exceeds that of the original processing element table, before the replacement, the contents of the elements whose names are greater than the name of the processing element must be transferred behind the memory address due to the excess in size. When the process for the processing element has been completed up to this stage, $C(JA)$ is written as zero (see Table 12).

Table 12 Processing table after the finished check

Reference Address	JA	1, 2, 3
Processing Element Name	$J(JA)$	2, 1, 0
Finished Check	$C(JA)$	0, 2, 0

Table 13 Processing table to be completed

Reference Address	JA	1, 2, 3
Processing Element Name	$J(JA)$	2, 1, 0
Finished Check	$C(JA)$	0, 0, 0

Fig. 11 Dividing process of element 1

Fig. 12 Dividing domain at completion of dividing process

3.9 Search for the processing element

The next processing element is sought in the processing table, and is processed according to the method outlined in sections 3.4 to 3.8.

In the present example, we examine $C(JA)$ in order of address. $C(JA)$ becomes nonzero when $JA = 2$ (see Table 12). Accordingly, a $J(2)$ value of 1 is assigned as the next processing element name. After the processes described in sections 3.4 to 3.8 have been carried out for element 1, the shared element as shown in Fig. 11 is obtained. Since the divided face, with the same name as the processing element 1, becomes one of the faces of the additional element, it is written in the additional element table (face address 2 in Table 15), and the divided faces with negative names such as -1 , -5 and -6 are copied into the frame table. Ordinarily, face name 2 of the remaining divided face should be registered in the processing table, since it is positive in sign, and is different from the boundary face name. However, since it is already registered, it is not written in this table again. When all the processes for element 1 have been completed, every $C(JA)$ becomes zero as in Table 13 and the dividing domain becomes that shown in Fig. 12 at this stage.

3.10 Composing process

When all the dividing processes have been completed, the boundary faces are all written in the additional element table. However, the composing process as mentioned in section 3.7.2 must be performed when the additional element has a frame face fragment. The number of frame face fragments with the frame face name of $-I$ has been memorized in the array variable $BC(I)$ in the frame table. Therefore, we search for the frame of $BC(I)=1$, and copy this fragment into the additional element table. Since $BC(4)$ is 1 in the present example, the frame face fragment in frame table -4 is copied into the additional element table (face address 3 in Table 15).

When $BC(I)$ is greater than or equal to 2, the frame face fragments in the frame table of $-I$ must be united into one face (composed face). The side (peripheral side) of the composed face is either a portion of the frame side or a newly cut line segment which resulted from the dividing process. When the new cutting line segment is produced, according to (1) of section 3.6.1, the top side name (top of the side name) in each face column of the shared element table always coincides with the frame face fragment name $BK(I, J)$. Thus, the condition (peripheral condition) which is required for the side to be the peripheral side, is that the side name is negative or the top side name coincides with $BK(I, J)$. The other sides (connecting sides) unite to form one composed face.

Table 15 Additional element table

Element Name N=3	Element Coordinates X(3),Y(3),Z(3)			Number of Faces AFN=6		
Face Address L	1	2	3	4	5	6
Face Name AF(L)	2	1	-4	-1	-5	-6
Number of Sides AEC(L)	4	4	4	4	4	4
Side Address M	1, 2, 3, 4	5, 6, 7, 8	9,10,11,12	13,14,15,16	17,18,19, 20	21,22,23,24
Side Name AE(M)	-6,-4,-5, 1	-6, 2,-5,-1	2,-6,-1,-5	1,-5,-4,-6	2,-4,-1, 1	2, 1,-1,-4

Table 14 Frame table (frame face -1)

Frame Face Name	-I	-1
Number of Fragment BC(I)		2
Face Address J	1	2
Face Fragment Name BK(I,J)	2	1
Number of Sides BEC(I,J)	4	4
Top Side Address BEI(I,J)	1	5
Side Address M	1, 2, 3, 4	5, 6, 7, 8
Side Name BE(I,M)	1,-5,-4,-6	1,-5, 2,-6

Since a side name of this face is also the name of another face linking with the face, there always exists a face column which has the frame face fragment name $BK(I, J)$ coincident with the connecting side name. Taking this into account, the composing process is conducted according to the flow chart shown in Fig. 13. Since $BC(1)$, $BC(5)$ and $BC(6)$ are all 2, the composing process must be performed for frames -1, -5 and -6. Let us consider the case of frame -1 as an example of the composing process, which will be explained in detail in the following section.

3.10.1 Search for the leading peripheral side

From the frame table, we determine the top side name which satisfies the peripheral condition and is called the leading peripheral side name. In the additional element table, we prepare the face column with the same name as the frame of the composed face, and write the leading peripheral side name there. In the present example, the frame table with the name -1 is shown in Table 14. Hence, the top side name of face address 2 satisfies the peripheral condition, and the leading peripheral side name 1 is written in the top side address ($M=13$) of face name -1 (face address 4) in the additional element table (Table 15).

3.10.2 Picking up the next side The next side is that of the next address, and the side next to the last side is the top side. In the example, the side next to the side 1 (side with the side name of 1) is side -5, in face 2 (face with the face name of 2) as shown in Table 14.

3.10.3 Examination of the peripheral side name If the side name is negative, and is equal to the name just written in the additional element table, the procedure goes back to section 3.10.2; the side name -5 has never been written until now.

3.10.4 Examination of the peripheral condition We examine whether or not the side name satisfies the peripheral condition. If the condition is not satisfied,

Fig. 13 Flow chart of the composing process

we proceed to section 3.10.7 since the side is the connecting side; the side name -5 satisfies the peripheral condition.

3.10.5 Examination of the leading peripheral side If the side name which satisfies the peripheral condition coincides with the leading peripheral side name, then the composing processes become complete; the side name -5 is not coincident with the leading peripheral side name.

3.10.6 Entry of the peripheral side name After entering the peripheral side name in the additional element table, we return to section 3.10.2. In the present example, the side name -5 is entered in the side address 14 in Table 15. Next, we return to section 3.10.2, and pick up the next side name 2. When examined according to sections 3.10.3 and 3.10.4, the next side is found not to satisfy the peripheral condition. Accordingly, the side name under consideration becomes the connecting face name. This name can be used to search for the frame face fragment to be linked to the face fragment under processing.

3.10.7 Search for the connecting face and side First, in the frame table, we search for the frame face fragment name $BK(I, J)$ coincident with the connecting face name. The connecting side name is the name of the face fragment which has been processed in sections 3.10.2 to 3.10.6. Next, in the side name item of that face fragment, we search for the side name $BE(I, M)$ coincident with the connecting side name. Finally, we return to section 3.10.2. In the present example, $BK(1, J)$ coincident with the connecting face

Fig. 14 Composing process
(frame face -1)

Fig. 15 Completed dividing domain

name 2 can be found at face address 1 in Table 14. $BE(1, M)$, coincident with connecting side name 1, can be found at side address 1 in Table 14, and we return to section 3.10.2.

Following these procedures, the processes described in sections 3.10.2 to 3.10.5 are carried out according to the flow chart as shown in Fig. 13 (see Fig. 14). When the series of composing processes are complete, the item of face address 4 in Table 14 is obtained.

The composing process in section 3.10 is performed for the frame faces -5 and -6. Thus, the additional element table of element 3 has been completed as shown in Table 15, and the dividing domain becomes that shown in Fig. 15.

3.11 Registration of the additional element table

This complete additional element table is added to the end of the element table. The efficiency of the process improves if extra memory space is secured for the array variables of the face and side names. When the element points were randomly located in the three-dimensional domain, the experience of the authors⁽⁹⁾ indicated that the Voronoi polyhedron had 15 faces and 5 sides per face on the statistical average. This experience is useful for determining the suitable extra memory size.

4. Conclusions

Since the automatic discretization method described in this report is carried out by adding the new elements one by one, it has slight disadvantages as follows: when an existing element point is taken off or moved, and when the shape of the dividing element has been given previously, the efficiency of the process is negatively affected. However, this method has many advantages and applications as outlined below.

(1) Since the automatic discretization is carried out merely by locating additional element points randomly, the shape of the body to be divided and the fineness of the divided elements can be chosen at will.

(2) The simplicity and ease of producing element tables facilitates the use of a microcomputer with a

small memory for the automatic discretization method.

(3) Owing to the characteristics of the Voronoi polyhedron, various types of partial differential equations can be easily transformed into simultaneous algebraic equations which can be processed in parallel accurately and quickly.

(4) The Voronoi polyhedron element table can be easily transformed into the usual tetrahedron element table.

(5) The Voronoi polyhedron element table has the potential for future growth, and can be used directly as an analytical method to obtain the numerical solution of the various partial differential equations.

References

- (1) Oda, J. and Yamazaki, K., The Present Condition of the Automatic Discretizing Method for the Finite Element and the Method of its Utilization, Kikai no kenkyu, (in Japanese), Vol. 37, No. 6 (1985), p. 704.
- (2) Green, P. J. and Sibson, R., Computing Dirichlet Tessellations in the Plane, Comput. J., Vol. 21, No. 2 (1978), p. 168.
- (3) For example, Ohya, T., Iri, M. and Murota, K., Improvements of the Incremental Method for the Voronoi Diagram with Computational Comparison of Various Algorithms, J. Oper. Res. Soc. Jpn., Vol. 27, No. 4 (1984), p. 306.
- (4) Bowyer, A., Computing Dirichlet Tessellations, Comput. J., Vol. 24, No. 2 (1981), p. 162.
- (5) Watson, D. F., Computing the n-Dimensional Delaunay Tessellation with Application to Voronoi Polytopes, Comput. J., Vol. 24, No. 2 (1981), p. 167.
- (6) Kobayashi, M., Maekawa, H. and Nakamura, H., Calculation of Mean Thermal Conductivity of Heterogeneous Solid Mixture by a Voronoi-Polygonal Element Method, 25th National Heat Transfer Symposium of Japan, (in Japanese), No. 1 (1988), p. 328.
- (7) Taniguchi, N., Arakawa, C. and Kobayashi, T., Construction of a Flow-Simulating Method with Finite Volume Based on Voronoi Diagrams, Trans. Jpn. Soc. Mech. Eng., (in Japanese), Vol. 55, No. 513, B (1989), p. 1324.
- (8) Kagaku Dai Jiten, (in Japanese), (1989), p. 1325, Maruzen.
- (9) Kobayashi, M., Maekawa, H., Nakamura, H. and Kondou, Y., Calculation of the Mean Thermal Conductivity of Heterogeneous Solid Mixture with the Voronoi-Polyhedron Element Method, Trans. Jpn. Soc. Mech. Eng., (in Japanese), Vol. 57, No. 537, B (1991), p. 1795. Translated in Heat Transfer Jpn. Res., to be published.