Numerical Heat Transfer, Part B, 64: 218–238, 2013

ISSN: 1040-7790 print/1521-0626 online
DOI: 10.1080/10407790.2013.797308

AN IMPROVED PAVING METHOD OF AUTOMATIC QUADRILATERAL MESH GENERATION

Yu Zhao[1], Bo Yu[1], and Wenquan Tao[2]
[1]*National Engineering Laboratory for Pipeline Safety, Beijing Key Laboratory of Urban Oil and Gas Distribution Technology, China University of Petroleum, Beijing, People's Republic of China*
[2]*MOE Key Laboratory of Thermal Fluid Science and Engineering, Xi'an Jiaotong University, Xi'an, People's Republic of China*

This article describes an improved paving method of automatic quadrilateral mesh generation. Paving, which was first proposed by Blacker and Stephenson [1], is a kind of direct method for generating a quadrilateral mesh and has been widely used since it was presented. This article aims to improve some weaknesses of the traditional paving method by generating high-quality quadrilateral grids without employment of a background mesh. Through efficient intersection resolution and other optimization measures, the improved paving method can generate well-aligned rows of quadrilateral elements almost parallel to the boundary of the domain, automatically and quickly.

1. INTRODUCTION

Numerical methods have been long recognized as powerful tools in the understanding and solution of complicated flow and heat transfer problems. The first and essential step that should be performed in these methods is discretization of the computational domain of the problem into a valid finite-element mesh. The size, shape, and number of elements in the mesh directly influence the final accuracy and cost of the numerical results [2, 3]. For most practical flow and heat transfer problems, which usually take place in irregular and complex domains, mesh generation of the domain is one of the biggest obstacles that must be overcome. Compared with a structured mesh, where all interior nodes have exactly four adjacent elements, an unstructured mesh is more suitable for discretization of irregular domains because of its topological flexibility and better control of size transition among elements. Since unstructured mesh generation has potentially a broad area of development, numerous research activities have been devoted to this topic. Quadrilateral meshes

Received 7 September 2012; accepted 18 March 2013.

The study is supported by the National Science Foundation of China (No. 51134006, No. 51176204, and No. 51276198), and the Science Foundation of China University of Petroleum, Beijing (No. 2462011LLYJ33, No. 2462011LLYJ55, and No. 2462012KYJJ0404).

Address correspondence to Bo Yu, Beijing Key Labarotary of Urban Oil and Gas Distribution Technology, China University of Petroleum, Beijing 102249, Peoples's Republic of China. E-mail: yubobox@vip.163.com

NOMENCLATURE

$d1$	distance between current paving edge and specific edge in the upstream, mm
$d2$	distance between current paving edge and specific edge in the downstream, mm
L	reference length of generating new edge based on size distribution of preexisting element on paving boundaries, mm
L'	reference length of generating new edge based on the distance between immediately preceding and subsequent boundary nodes, mm
$L1$	length of specific edge in the upstream, mm
$L2$	length of specific edge in the downstream, mm
N	element node
Q_{EAS}	equiangle skewness
α	internal angle of node, deg
Subscripts	
i, j, k, l, m, n	number of element nodes shown in Figure 2

and triangular meshes are the most widely used meshes in two-dimensional mesh generation. At this point a fully automatic generation technique for triangular elements has already been established and developed perfectly. On the other hand, automatic mesh generation of quadrilateral elements still has room for improvement, and a number of researchers [4–6] point out the fact that a high-quality quadrilateral mesh can give better solution than a triangular mesh with a similar number of elements. Masud and Khurram [7] confirmed the superiority of the quadrilateral mesh in computational accuracy by comparing the numerical errors of convection-diffusion equation with one of triangular meshes, and their research also verified that the convergence rate of numerical computation in the case of quadrilaterals was faster than with triangles. Therefore, most research on creating fully automatic unstructured mesh generators focuses on the automatic generation of quadrilateral element meshes in arbitrary geometries.

Since the 1980s, unstructured quadrilateral mesh generation technology has developed rapidly, and extensive methods have already been proposed by researchers around the world. Generally speaking, these methods can be divided into two main categories: indirect and direct methods.

1.1. Indirect Methods

With an indirect method, the computational domain is first meshed with triangles and then various algorithms are employed to convert the triangles into quadrilaterals. A simple conversion method was proposed by Lo [8] using selective removal of diagonals between triangles in order to maximize the number of quadrilaterals. However, this method suffered from mixed types of triangular and quadrilateral elements. Later on, Lee and Lo [9] developed an enhancement of Lo's method by employing local triangle splitting and swapping. A similar method was proposed by Johnston et al. [10] for converting all the triangles in a triangular mesh into quadrilaterals. In addition, an advancing-front method which performs the merging process simultaneously with advancement of fronts was developed by Zhu et al. [11]. Recently, Petersen et al. [12] proposed a grid-based indirect approach with graded quadrilateral meshes, while Merhof et al. [13] presented another new indirect approach based on discrete surfaces.

With indirect methods, quadrilateral elements are generated from the background triangular mesh, so it is not difficult to achieve a high element density gradient, and complicated irregular domains can also be handled easily with a suitable triangular mesh-generation method. Unfortunately, the elements generated with indirect methods are usually of lower quality compared to those generated by direct methods.

1.2. Direct Methods

With a direct method, quadrilateral elements are placed in the domain directly, without triangular mesh generation beforehand. Many direct methods of unstructured quadrilateral mesh generation have been presented in the literature, and they can be grouped into two main categories. The first is known as the domain decomposition method (DDM) and depends on some form of decomposition processes to decompose the original domain into simpler suitable or mappable regions. Methods of this category were proposed by Baehmann et al. [14], Talbert and Parkinson [15], Tam and Armstrong [16], Joe [17], and other researchers. Because these methods achieve the purpose of mesh generation through regional decomposition, they can avoid large-scale geometric calculation so the meshing speed is quite desirable, but they are difficult to automate and elements near boundaries are of generally poor quality.

The second category consists of methods that utilize an advancing–front method (AFM) for mesh generation. From nodes which are initially placed on permanent boundaries, individual elements are sequentially formed and then extended to the internal region until the entire computational domain is fully filled by quadrilateral elements. Application of AFMs can be dated back to the work of Zhu et al. [11], while Blacker and Stephenson [1] proposed a direct advancing–front method called "paving", where complete rows of quadrilateral elements are generated directly, starting from the permanent external or internal boundary and pushing forward to the interior of the domain.

Of the methods discussed above, the paving method provides some desirable features consistently. As Blacker and Stephenson described in their literature, the final quadrilateral mesh generated with the paving method is boundary-sensitive, which means that mesh contours closely follow the contours of the boundary, and the elements near the boundary are usually well shaped. In addition, through carefully controlled process and various clean-up and smoothing measures, few irregular nodes are maintained, and rotating or translating the given domain will not change the resulting mesh topology [1]. Thus many researchers prefer to employ paving as an automatic quadrilateral mesh generater rather than the other techniques for these three reasons.

However, some disadvantages of the paving method also need to be addressed. Lacking fundamental support of mathematics theory, mesh generation with the paving method is faced with the intersection problem frequently. Detection and resolution of intersections usually costs a lot of time, since all the edges on the paving boundary should be checked at the step of new edge generation. How to control the size of a single element in order to smooth the size transition of the whole mesh is another problem that often occurs with the paving method, especially when boundaries contain elements of greatly differing size. In conclusion, the paving method cannot control the element edge generation by the whole distribution of step size and cannot localize the intersection check. Although many researchers have developed their own methods and insisted on their robustness and advantages, most of the current methods have limitations, and a reliable method of generating a high-quality quadrilateral mesh is still challenging [18, 19].

This article proposes an improved paving method of automatic quadrilateral mesh generation which contains a series of effective measures different from those presented by former researchers. Focusing on the two main unsatisfying aspects of the paving method, the improvements aim to not only maintain the desirable features of the traditional paving method, but achieve smooth size transition of the whole mesh and efficient detection of intersections.

2. OVERVIEW OF THE IMPROVED PAVING METHOD

Successful implementation of the improved paving method involves a series of tightly controlled procedures to ensure mesh validity and quality. Though many of the operations come from the traditional paving method and subsequent research, it is still necessary to briefly outlined the whole process of mesh generation in the following steps:

1. *Discretization of the permanent boundary*. Nodes are carefully placed on the permanent boundaries based on the initial geometric information.
2. *Paving edge classification and choice*. As can be seen in Figure 1, every edge on the paving boundary is first sorted according to its state, as introduced by Owen et al. in their Q-Morph method [20]. The state of a paving edge is determined by computing the angle at the nodes on either end of the edge with each of its adjacent paving edges and will determine how the paving edge is going to the used in the process of mesh generation.
3. *Element generation*. Before a new element is generated, new node(s) must be placed from the nodes on the paving edge in different ways, depending on the internal angle α, as shown in Figure 2, and then edges are formed by connecting nodes to generate a new element eventually.

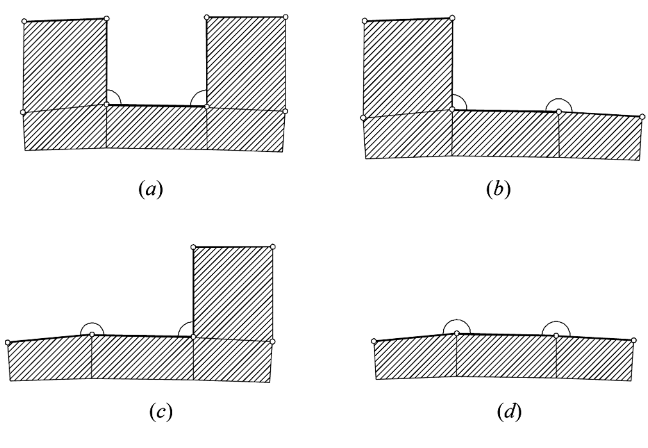


Figure 1. States of paving edge. (*a*) state 1–1; (*b*) state 1–0; (*c*) state 0–1; (*d*) state 0–0.

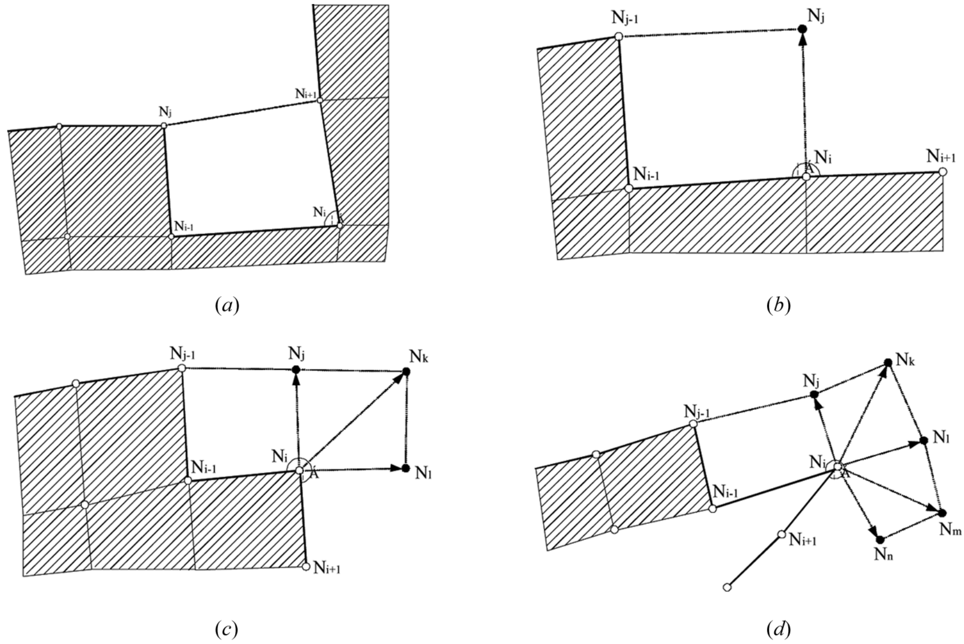


Figure 2. Classification of paving nodes. (*a*) End node; (*b*) side node; (*c*) corner node; (*d*) reversal node.

4. *Detection and resolution of intersections*. After a new edge is generated, it must be checked to find out whether it intersects with other edges on the paving boundaries. If there is an intersection, some sophisticated measures must be immediately taken to guarantee the desirable quality of local elements.
5. *Pre-adjustment*. Before generation of a single new element and after one row of elements is completed, pre-adjustment measures are used to deal with small angle or abnormal length ratio between two adjacent paving edges to avoid severe distortion of the element.
6. *Closure check*. A procedure is designed to check whether there is no available paving edge on the current paving boundary. If so, specific closure operations are employed to finish the current paving boundary.
7. *Topological clean-up*. When the entire mesh generation is finished, topological clean-up measures, such as insertion or deletion of elements, are used to improve the overall mesh quality.
8. *Smoothing*. The smoothing operation is widely used not only after resolution of intersections, per-adjustment, or when one paving boundary is terminated, but after the whole domain is completely meshed, to further improve the element quality.

The improved paving method is an iterative process which uses the critical steps introduced above. Some of these procedures have been well developed and presented previously in the literature, so this article focuses mainly on the aspects which still have room for improvement, such as better control of element size, efficient detection and resolution of intersections, optimization of pre-adjustments, and topological clean-up. All these improvements will be described in detail later on.

3. IMPLEMENTATION OF IMPROVEMENTS

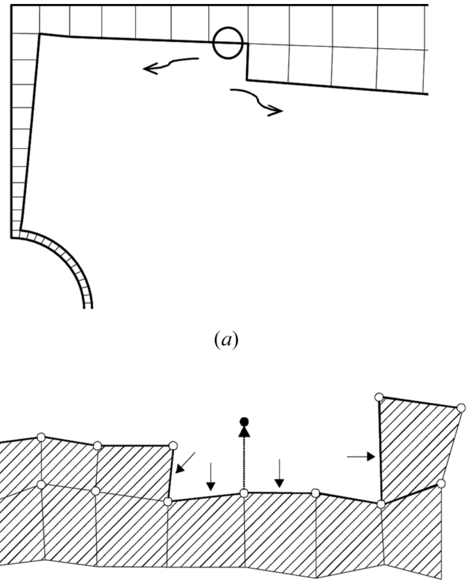
3.1. Element Size Control

Extensive research has put emphasis on element size control to make sure that the final quadrilateral mesh has a uniform size distribution. The quad-morphing (Q-Morph) method [20] proposed by Owen et al. takes advantage of local topology information from the initial triangular background mesh and utilizes an advancing-front algorithm to convert a triangular mesh into an all-quadrilateral mesh. Through sophisticated procedures to control the process of merging two triangles into a single quadrilateral, the final mesh has a desirable overall size distribution and elements near the boundary are also well shaped. However, the complexity of the Q-Morph method itself and alternate use of triangular and quadrilateral meshes may lead to complicated data structure and instability of the program, which also occur in some direct methods that rely on a background mesh to control the size of elements. Methods proposed by Lo [8] and Cheng et al. [21] can generate a quadrilateral mesh where overall size transition is very smooth but many irregular nodes are maintained eventually. Not employing a background mesh, Garimella et al. [22] and Chen et al. [23] proposed a method using a "local coordinate system" and "parameter space", respectively, to control the size of elements and extended it to unstructured quadrilateral mesh generation for a 3-D curved surface. Since a large number of calculations is necessary when every new element is formed, this method suffers from a low efficiency of working. Park et al. [24] came up with inserting virtual nodes to control the size of elements, but insertions are only implemented in the permanent boundary. As a result, when elements are generated in the interior of the domain, size transition is not under control.

The improved paving method controls the size of generated elements through comprehensive consideration of the size distribution of preexiting elements on current and other paving boundaries instead of adopting the background mesh or using other methods which need a long time to work. Although similar to Blacker and Stephenson's method in some aspects, the improved paving method utilizes additional features that help generate high-quality elements near the boundary and control the element edge generation by the whole distribution of mesh size.

3.1.1. Size distribution on the current paving boundary. The first step of element generation is placing new nodes properly from paving nodes on the current paving boundary to form a new edge. As shown in Figure 2, each paving node should be classified according to the size of the node's interior angle, α, and some settings of angle tolerances. All paving nodes can be eventually classified as end ($0° < \alpha \leq 135°$), side ($135° < \alpha \leq 225°$), corner ($225° < \alpha \leq 315°$), or reversal nodes ($315° < \alpha \leq 360°$). For end paving nodes, a new element is simply formed without generating any new node. But for the other three categories of paving nodes, before formation of an element, new nodes and edges must be carefully placed in order to control the ratio of size between different elements.

Taking the size distribution of the current paving boundary into consideration, as can be seen in Figure 3, a procedure is introduced to search along the upstream and downstream of the current paving boundary to find a specific edge which contains an end node, and the size of the specific edge will influence the generation of a new edge in the future.



(*a*)

(*b*)

Figure 3. Element size control on the same paving boundary. (*a*) Search for specific edge containing end node; (*b*) comprehensive consideration of pre-existing elements sizes.

When two such edges are caught in the upstream and downstream, lengths of the two edges (let them be $L1$ and $L2$, respectively) and distances between the current paving edge and each specific edge (let them be $d1$ and $d2$, respectively) will all be recorded. Then the reference length L can be obtained

$$L = L1 \times \frac{d2}{d1 + d2} + L2 \times \frac{d1}{d1 + d2} \tag{1}$$

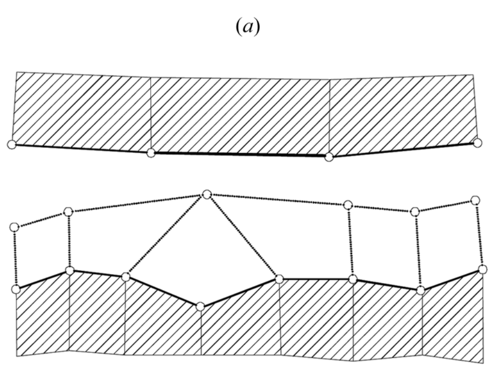
Just like gravitation in physics, the closer the specific edge is to the current paving edge, the greater is its influence on the determination of the length of the new edge. So, bringing a weighted factor to control the size of elements can make the existing size distribution transfer to the interior of the domain smoothly. In addition, according to the traditional paving method, just based on the distance between immediately preceding and subsequent boundary nodes as well as the node classification, another reference length L' can be obtained. Then the comparison between L and L' is implemented to check whether $L/L' > 1.5$, or $L'/L > 1.5$, which means if generation of the new edge is determined by L or L' separately, the ratio of adjacent edges in the same element may be undesirable and the transition among these mesh elements is not so graceful. So if the situation where $L/L' > 1.5$ or $L'/L > 1.5$ is detected in the process of generating a new edge, the length of the new edge relies on the arithmetic average of both L and L'; otherwise it is determined by the reference length L_0.

Most paving boundaries contain at least two or more end nodes. However, there are still some special cases of paving boundaries that do not contain as end node initially and need to be handled differently. That is, if no specific edge is found after going through the entire paving boundary, a new edge is generated based only on the classification of the current paving node and the distances between it and the other two adjacent nodes.

3.1.2. Size distribution on other paving boundaries. Different from simple connected domains, for multiple connected domains generation of a new element based only on the preexisting elements on the current paving boundary while not considering the size distribution of other boundaries may result in poor element quality when different boundaries meet each other.

The improved paving method generates a new element under the comprehensive consideration of size distribution of different paving boundaries. The specific way is to record the average size of all elements in each boundary respectively and compare the size of the new generated element with the recorded size of the opposing boundary. If the size of the current element turns out to be 1.5 times greater than that of the opposing boundary, the generation process of the current paving boundary is paused and other boundaries continue to generate new elements to smooth out the size diversity between elements in different boundaries.

However, the operation discussed above can only control the distribution of element size on a large scale; when the difference of element size is great though the distance between different paving boundaries is relatively small, more effective and timely measures should be taken. As shown in Figure 4*a*, the element



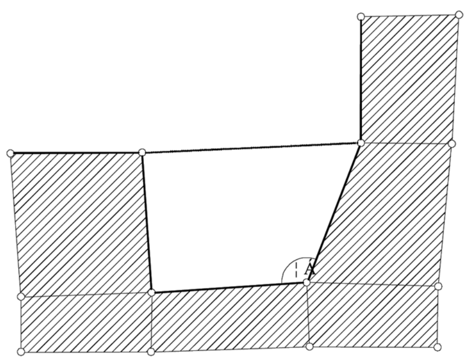
(*a*)

(*b*)

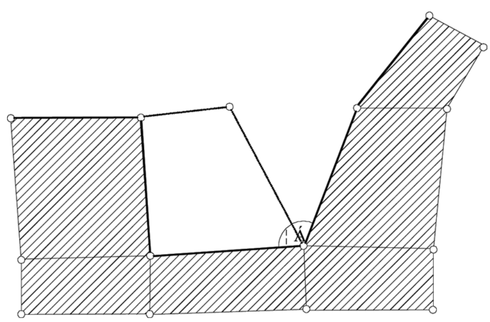
Figure 4. Element size control on different paving boundaries. (*a*) Large size difference of different paving boundaries; (*b*) insertion of special element.

size of the two opposing boundaries differs greatly, so when such an undesirable situation is detected, the special element illustrated in Figure 4*b* is automatically inserted to reduce the difference and guarantee thus the process of mesh generation continues smoothly. It is also worth pointing out that if it is less than absolutely necessary, the special element will not be inserted, in order to keep irregular nodes away from the permanent boundary as far as possible.

3.1.3. Optimized closure. With an end node, which is often encountered around a corner of a boundary, operation of the traditional paving method forms a single element simply, without any node projection and edge generation. The improved paving method deals with the end node according to the circumstances around it. When the angle of an end node is between 100° and 135° as shown in Figure 5*a*, that means the end node is quite close to a side node, so if the simple operation of the traditional method could result in formation of a new end node on the next paving edge, the direct connection is taken. However, if such a simple operation will make the state of the next paving edge become 0-0 as shown in Figure 5*b*, the current end node is treated as a side node to generate the new node and edge.



(*a*)



(*b*)

Figure 5. Optimized treatment of end node. (*a*) Simple closure; (*b*) generation of new node and paving edge.

3.2. Detection and Resolution of Intersection

3.2.1. Local detection of intersections. The traditional paving method generates elements row by row starting from the permanent boundary and working toward the interior. White and Kinney [25] proposed enhancements to paving, suggesting individual placement of elements rather than complete rows. No matter what measures are taken to generate a single element, all edges on all the paving boundaries must be checked. This process is a global check but not a local check. As a result, detection of intersections is the most time-consuming step among all the steps of the paving method, and the speed of mesh generation is slow. Q-Morph, as proposed by Owen et al., can localize the intersection check, but it has the risk of changing the distribution of the background mesh, which may lead to failure of the size space and poor quality of the final quadrilateral mesh.

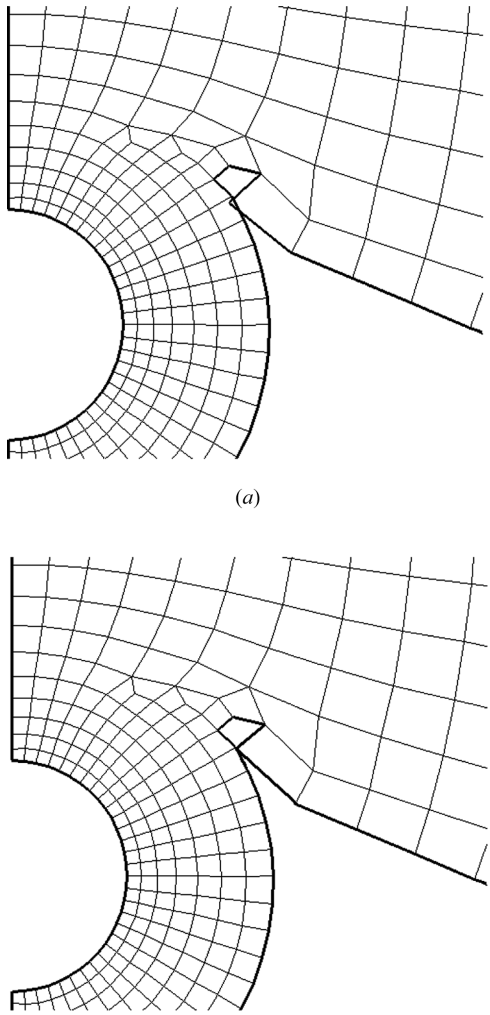
When a paving boundary intersects with itself, the position of the intersection is usually in the corner of the boundary, that is, the new edge of the element always intersects with other edges which are not far away from it. The improved paving method takes some measures which are similar to dichotomy, a classical mathematical algorithm, to localize the intersection check. Before the step of intersection begins, let the current paving edge be the midpoint so the whole paving boundary is divided into two parts, an upstream part and a downstream part. Then, starting from the edge of the upstream part which is nearest to the current paving edge, all edges of the upstream part are checked, and if no intersection appears, rather to the current paving edge again and check the other part of the paving boundary, the downstream part. About half of time can be saved in this way, especially for complicated domains which may come across large numbers of intersection problems.

In order to go back to the initial paving edge quickly after going through all edges of the upstream part of boundary, a "pointer chain" is employed in the program, linking edges on the paving boundary end to end to speed up the process of intersection check. In addition, introducing a "pointer" to program can also better adapt to the situation of adding or deleting nodes and elements occurring occasionally in the process of mesh generation.

In the program, "structure" data type is also used to integrate essential information of each edge to build up the whole understanding of the object and reduce the complexity of the program.

3.2.2. Efficient resolution of intersections. Resolution of intersections usually involves merging or splitting of the paving boundary, and in this process a number of edges in different boundaries and topological information about nodes and edges all need to be recalculated. After these updates, definition and reselection of paving boundaries also need a large amount of calculation.

As presented in section 3.2.1, the intersection position when a paving boundary crosses itself is always in a corner, but for this situation most algorithms still go through the general way, which consumes a out of time. The improved paving method checks the number of edges between the current paving edge and the edge it intersects with, and if the number is three (not including the current paving edge itself), as shown in Figure 6, a single quadrilateral element is formed simply so that complex merging or splitting of boundaries can be avoided, enhancing the efficiency of program.



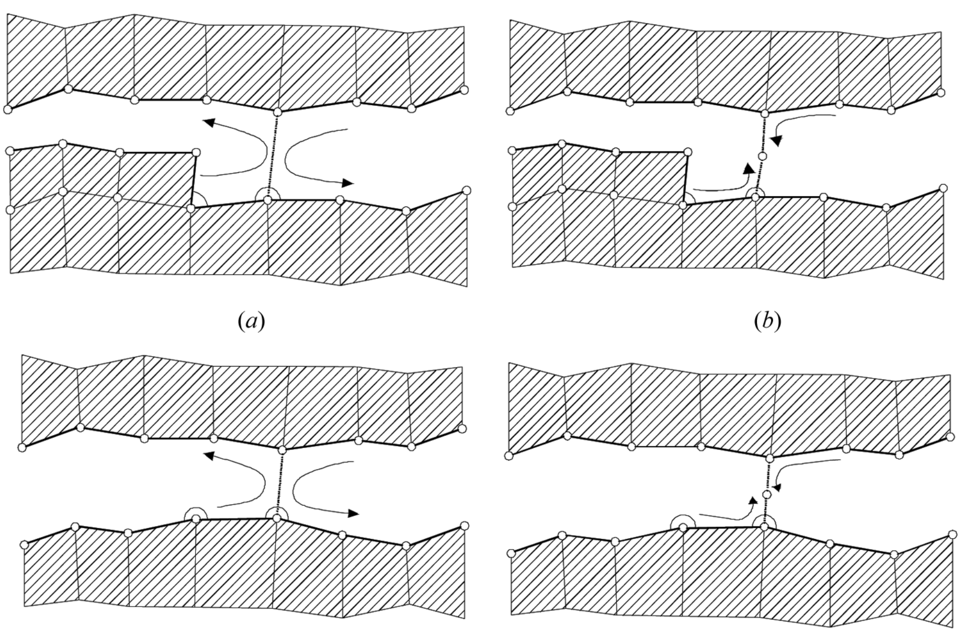
(*a*)

(*b*)

Figure 6. Efficient resolution of special intersection. (*a*) Intersection around corner of paving boundary; (*b*) direct closure with single element.

When it comes to the situation that merging or splitting of boundaries is inevitable, common treatments in most literature are to select and form new edge(s) according to the potential number of paving edges on the new paving boundary. In order to form an all-quadrilateral mesh, every paving boundary must contain an even number of edges, so the potential number of edges is first calculated. If the number is even, as in Figures 7*a* and 7*c*, then a new edge is made by connecting nodes on the two opposing edges to define a new paving boundary. On the other hand, if the number of edges on the potential boundary is odd, a new node is generated first instead of the direct connection, as shown in Figures 7*b* and 7*d*, to guarantee that each paving boundary has an odd number of edges, and then two edges are formed to close the new paving boundary. Such operations are widely used in resolution of intersections, but, the different paving direction on opposite sides of new edge(s) may result in instability and complexity of program structure, and generation of new nodes to maintain an even boundary may also lead to great length-to-width ratio when new elements are generated near the intersection position.

In the improved paving method, a single quadrilateral element is formed directly to merge or split the paving boundary. In the case in Figure 8*a*, three adjacent nodes on the current paving boundary and one node on the opposite boundary are connected to create a new element. In another case, shown in Figure 8*b*, two adjacent nodes on the current paving boundary and two other nodes are connected to generate a new element. Similar to traditional methods, the potential number of paving edges after assumed connection is calculated first, to avoid the occurrence of an odd boundary. The difference is that if the potential number turns out to be odd, the new element will be generated in another position and the final position is determined by evaluating and comparing the quality of the element when it is hypothetically generated in each of the two optional positions as shown in Figures 8*c* and 8*d*. Then the new element which aims to merge or split different boundaries is generated eventually in the position where the quality of the element is more desirable than that of another position.



(*a*) (*b*) (*c*) (*d*)

Figure 7. Traditional treatments of general intersection with state 1–0 paving edge or state 0–0 paving edge. (*a*) State 1–0 paving edge, even number; (*b*) state 1–0 paving edge, odd number; (*c*) state 0–0 paving edge, even number; (*d*) state 0–0 paving edge, odd number.

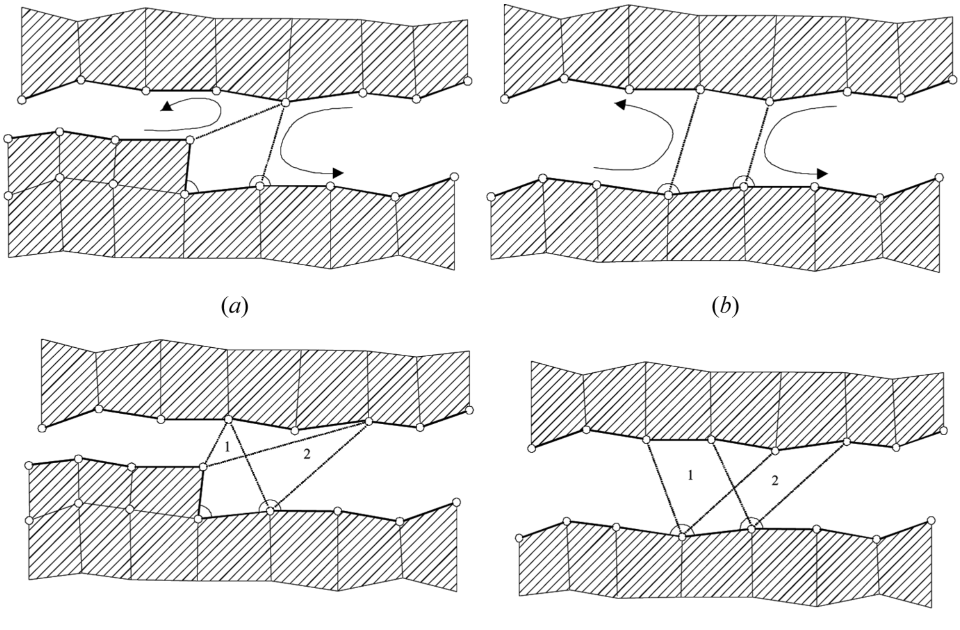


Figure 8. Treatments of general intersection in the improved paving method when the potential number of paving edges is even or odd. (*a*) State 1–0 paving edge, even number; (*b*) state 0–0 paving edge, even number; (*c*) state 1–0 paving edge, odd number; (*d*) state 0–0 paving edge, odd number.

3.3. Pre-adjustment and Topological Clean-Up

Though the whole process of quadrilateral mesh generation is under careful control, pre-adjustment for the paving boundary is very necessary when the node's interior angle on the current paving boundary tends to become small or the length ratio between the neighboring edges on the current paving boundary tends to become large due to the large gradation of element sizes and the evolution of the paving boundary. The measures of pre-adjustment employed in this article include three main categories as the traditional paving method presented: seaming, transition seaming, and transplitting seaming.

After the computational domain has been completely paved, local topological clean-up measures are taken to improve the overall quality of the finished mesh. These measures include insertion or deletion of elements with poor interior angles to reduce the number of irregular nodes and an attempt to make most internal nodes connected with four elements. The main clean-up measures [26] used in the improved paving method are (1) delete the isolated node; (2) merge specific nodes; (3) 4-2 transform; and (4) insert an element when an internal node has six adjacent nodes. Combined with these clean-up measures, the program iteratively uses local smoothing to reduce the number of irregular nodes as much as possible. Finally, the overall Laplacian smoothing and an optimization-based smoothing operation [26] are implemented to improve the mesh.

4. MESH EXAMPLES AND PERFORMANCE RESULTS

4.1. Mesh Examples

Based on the improved paving method in this article, an automatic quadrilateral mesh generator was programmed in Fortran95 language to discretize some irregular domains (as shown in Figures 9–12). For the exactly same domains where the positions of nodes on the permanent boundary are identical to that of the improved paving method, all-quad meshes are also generated by the well–known computational fluid dynamics (CFD) software package Gambit, which is designed to generate a quadrilateral mesh using a paving method of its own. The display of

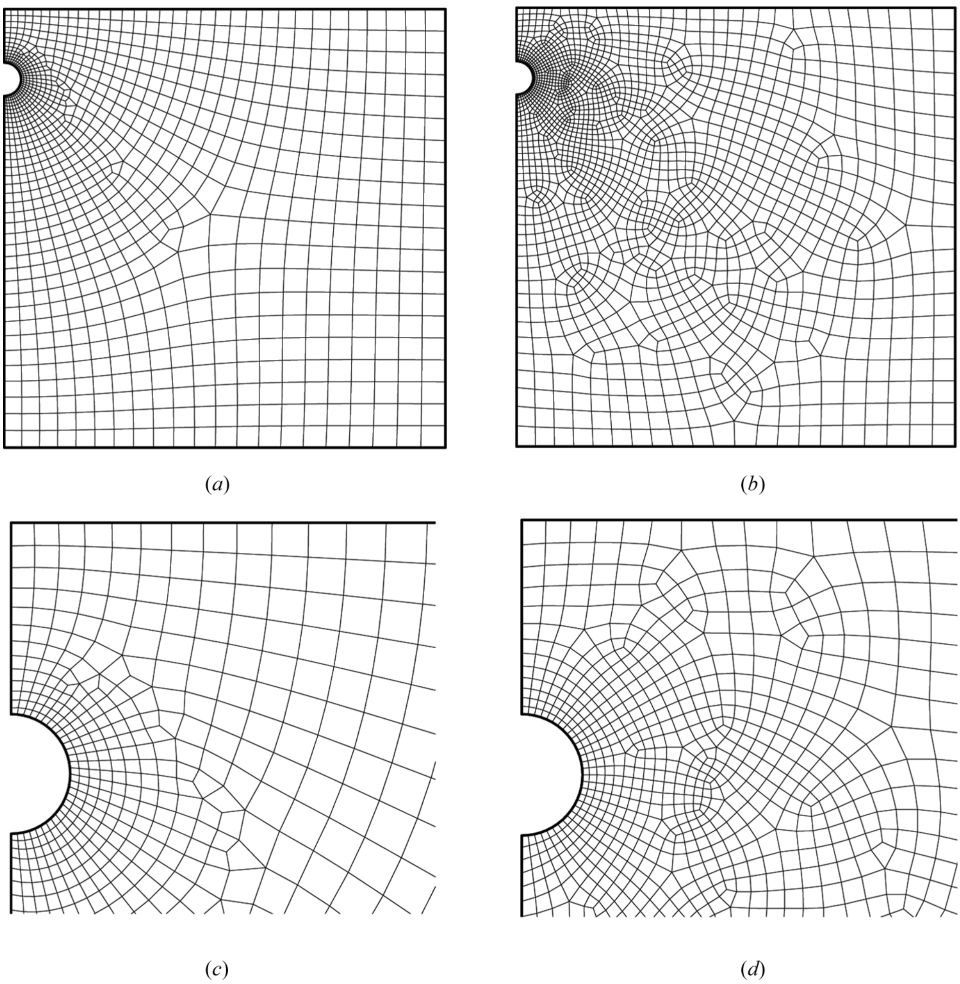
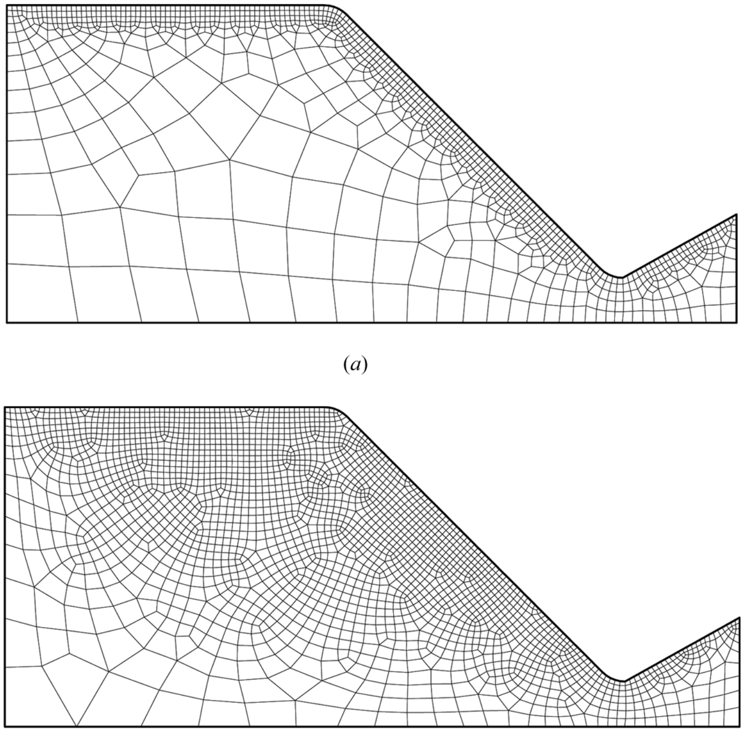


Figure 9. Comparison of the improved paving method and Gambit in a semicircular pit domain. (*a*) The improved paving method; (*b*) Gambit; (*c*) elements around semicircular pit of (*a*); (*d*) elements around semicircular pit of (*b*).



(*a*)

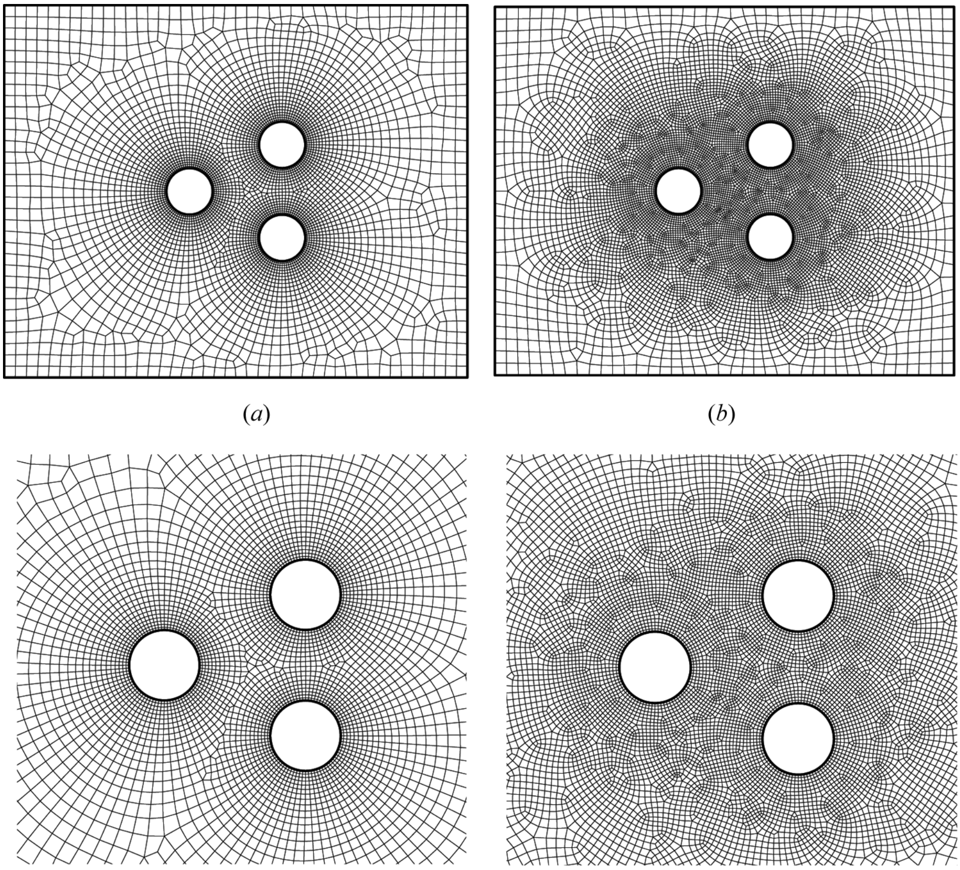
(*b*)

Figure 10. Comparison of the improved paving method and Gambit in a large-transition domain. (*a*) The improved paving method; (*b*) Gambit.

these mesh examples and elements distribution around boundaries demonstrates various features of the improved paving method.

The first example, shown in Figure 9, consists of a square exterior permanent boundary which contains a semicircular pit on one side of it. The exterior permanent boundary has a nonuniform spacing of nodes. Nodes on the pit are approximately twice as dense as on other portions of the boundary. Compared with Figures 9*a* and 9*b*, elements generated by the improved paving method are aligned better to form clear rows with fewer irregular nodes, which is depicted more clearly in Figures 9*c* and 9*d*.

Figure 10 shows the meshing result of an irregular domain requiring a high degree of transition. The density of nodes near the top of the permanent boundary is much higher than that of nodes on other parts of the permanent boundary, especially the bottom corners. The mesh depicted in Figure 10*a* illustrates that the improved paving method can maintain the desired mesh density for future CFD or numerical heat transfer (NHT) applications while still enforcing well-aligned rows of elements transitioning quickly to larger-size elements. On the other hand, the mesh shown in Figure 10*b* contains too many elements, which will greatly increase the number of computer operations.



(*a*) (*b*)

(*c*) (*d*)

Figure 11. Comparison of the improved paving method and Gambit in a three-circle multi-connected domain. (*a*) The improved paving method; (*b*) Gambit; (*c*) elements around three circles of (*a*); (*d*) elements around three circles of (*b*)

The multi-connected domain of Figure 11 is composed of a rectangle exterior permanent boundary and three circular interior permanent boundaries which have the same diameter. Nodes on the three circles are about three times as dense as that on the exterior boundary. Figure 11*c* further illustrates the ability of the improved paving method to generate well-aligned rows of elements, while still maintaining the required element size transitions. As illustrated in Figure 11*d*, the method Gambit employs introduces many irregular internal nodes and has difficulty in forming well-aligned rows of elements. Furthermore, as shown in Figure 12, the ability to generate high-quality quadrilateral elements while accomplishing quick size transitions of different boundaries is depicted better with the comparison of meshing results of a more complicated multi-connected domain where the density of nodes on several permanent boundaries containing arcs or circles is higher than other parts of the domain.

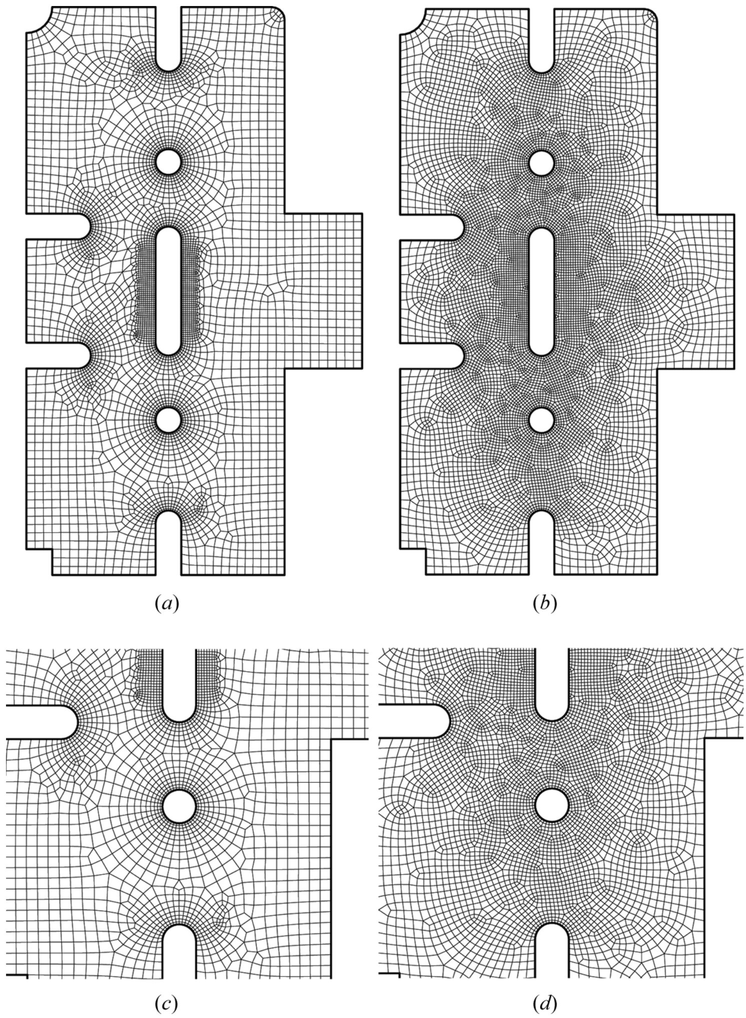


Figure 12. Comparison of the improved paving method and Gambit in a complicated multi-connected domain. (*a*) The improved Paving method; (*b*) Gambit; (*c*) elements around the bottom of (a); (*d*) elements around the bottom of (*b*).

4.2. Performance Results

4.2.1. Mesh quality. As can be seen in Figures 9–12, the improved paving method and Gambit software generate different meshes for the several same domains. To evaluate element quality, skewness of the mesh is calculated based on a normalized measure known as EquiAngle (Q_{EAS}) [27] as

$$Q_{EAS} = \max\left[\frac{\theta_{max} - \theta_e}{180 - \theta_e}, \frac{\theta_e - \theta_{min}}{\theta_e}\right] \quad (2)$$

where θ_{max} and θ_{min} are the maximum and minimum angels in degrees between the edges of the element and θ_e is the characteristic angle corresponding to an equilateral

Table 1. Comparison of mesh quality of the improved paving method and Gambit

Domain	Method	Number of quadrilaterals and proportion of the total						Total quads
		$Q_{EAS} =$ 0–0.1	$Q_{EAS} =$ 0.1–0.2	$Q_{EAS} =$ 0.2–0.3	$Q_{EAS} =$ 0.3–0.4	$Q_{EAS} =$ 0.4–0.5	$Q_{EAS} =$ 0.5–1	
Semicircular pit (Figure 9)	IPaving	775 77.89%	174 17.49%	28 2.81%	9 0.90%	9 0.90%	0	995
	GPaving	845 56.15%	400 26.58%	122 8.11%	128 8.50%	9 0.60%	1 0.07%	1505
Large transition (Figure 10)	IPaving	479 50.26%	269 28.23%	126 13.22%	65 6.82%	10 1.05%	4 0.42%	953
	GPaving	1575 64.92%	428 17.64%	191 7.87%	205 8.45%	24 0.99%	3 0.12%	2426
Three-circle (Figure 11)	IPaving	4,015 90.12%	300 6.73%	82 1.84%	44 0.99%	14 0.31%	0	4455
	GPaving	6,857 68.01%	1,818 18.03%	599 5.94%	803 7.96%	5 0.05%	0	10082
Complicated multi-connected (Figure 12)	IPaving	3,077 68.94%	871 19.52%	355 7.95%	117 2.62%	43 0.96%	0	4463
	GPaving	5,991 61.67%	2,139 22.02%	747 7.69%	825 8.49%	13 0.13%	0	9715

cell of similar form, which would be 90° for a quadrilateral element. By definition, therefore, Q_{EAS} is between 0 (square element) and 1 (degenerate element). Usually the maximum Q_{EAS} must below 0.9 and the majority of elements must have skewness below 0.5.

The evaluation results of mesh quality of the improved paving method and Gambit software (let them be IPaving and GPaving, respectively) are shown in Table 1. From the comparison, more than 50% of the total quadrilaterals in each of these mesh examples generated by the improved paving method are of very good quality ($Q_{EAS} = 0–0.1$); meanwhile, very few quadrilateral elements suffer from severe distortion ($Q_{EAS} > 0.5$). For the semicircular pit and complicated multi-connected domain, the proportions of elements which are close to square elements are 77.89% and 68.94%, respectively, compared to the 56.15% and 61.67% of Gambit. Especially for the three-circle domain, as much as 90% of the quadrilaterals generated by the improved paving method are well shaped, even though the total number of elements is much smaller than that of Gambit. For the large-transition domain, though the proportion of high-quality elements is smaller than that of Gambit, the total number of quadrilateral elements of the improved paving method is only one-third of that of Gambit and, as shown in Figure 10, the quality of elements near boundaries is still quite desirable.

In addition, it is worth pointing out that since Gambit does not have any option to obtain a certain number of quadrilateral elements, the numerical tests mentioned above could not be carried out with a similar number of elements on the IPaving and GPaving methods. And it seems more common and convenient to generate a computational mesh in engineering practice only with the geometry information and initial nodes distribution of the computational domain, rather than controlling the number of generated grids.

Table 2. Meshing speed of the improved paving method

Domain	Method	Total quads	CPU time (s)
Semicircular pit	IPaving	995	0.25
Large transition	IPaving	953	0.36
Three-circle	IPaving	4,455	13.65
Complicated multi-connected	IPaving	4,463	16.15

4.2.2. Meshing speed. The improved paving method speeds up the detection and resolution of intersections by introducing efficient check produces and a "pointer chain" to the program. Table 2 shows meshing speed of the improved paving method in the form of CPU time. All tests were performed on a 2.83-GHz computer.

5. CONCLUSIONS

The improved paving method is a direct quadrilateral meshing method that utilizes an advancing-front approach to generate quadrilateral elements in the computational domain. It borrows many techniques from the traditional paving method proposed by Blacker and Stephenson but resolves some of its inherent problems by developing various measures different from those of existing methods. Depending on the existing elements to control the size distribution, the whole mesh achieves smooth size transition without generation of a background mesh beforehand, common to most indirect methods of advancing-front meshing. Improvements also include efficient detection and resolution of intersections by employing measures which speed up the process of intersection check and optimize the merging and splitting of paving boundaries. And various facilities for pre-adjustment and topological clean-up have been also addressed throughout the entire process of mesh generation.

The resulting meshes of the improved paving method contain few irregular internal nodes and high-quality elements whose contours, in general, follow the boundary of the domain. Overall mesh quality is pretty desirable, at the same time as the number of elements is reasonably small to reduce the number of computer operations in future numerical calculations.

REFERENCES

1. T. D. Blacker and M. B. Stephenson, Paving: A New Approach To Automated Quadrilateral Mesh Generation, *Int. J. Numer. Meth. Eng.*, vol. 32, pp. 811–847, 1991.
2. W. Q. Tao, *Numerical Heat Transfer*, 2nd ed., pp. 434–435, Xi'an Jiaotong University Press, Xi'an, China, 2002.
3. T. J. Baker, Mesh Generation: Art or Science? *Prog. Aerospace Sci.*, vol. 41, pp. 29–63, 2005.
4. J. Z. Zhu, E. Hintorc, and O. C. Zienkiewicz, Adaptive Finite Element Analysis with Quadrilaterals, *Comput. Struct.*, vol. 40, pp. 1097–1104, 1991.
5. S. H. Lo and C. K. Lee, On Using Mesh of Mixed Element Types in Adaptive Finite Element Analysis, *Finite Elem. Anal. Des.*, vol. 11, pp. 307–336, 1992.

6. P. Kraft, Automated Remeshing with Hexahedral Elements: Problems, Solutions and Applications, *Proc. 8th Int. Meshing Roundtable*, pp. 357–367, 1999.
7. A. Masud and R. A. Khurram, A Multiscale/Stabilized Finite Element Method for the Advection–Diffusion Equation, *Comput. Meth. Appl. Mech. Eng.*, vol. 193, pp. 1997–2018, 2004.
8. S. H. Lo, Generating Quadrilateral Elements on Plane and over Curved Surfaces, *Comput. Struct.*, vol. 31, pp. 421–426, 1989.
9. C. K. Lee and S. H. Lo, A New Scheme for the Generation of a Graded Quadrilateral Mesh, *Comput. Struct.*, vol. 52, pp. 847–857, 1994.
10. B. P. Johnston, J. M. Sullivan, and A. Kwasnik, Automatic Conversion of Triangular Finite Element Meshes to Quadrilateral Elements, *Int. J. Numer. Meth. Eng.*, vol. 31, pp. 67–84, 1991.
11. J. Z. Zhu, O. C. Zienkiewicz, E. Hinton, and J. Wu, A New Approach to the Development of Automatic Quadrilateral Mesh Generation, *Int. J. Numer. Meth. Eng.*, vol. 32, pp. 849–866, 1991.
12. S. B. Petersen, J. M. C. Rodrigues, and P. A. F. Martins, Automatic Generation of Quadrilateral Meshes for the Finite Element Analysis of Metal Forming Processes, *Finite Elem. Anal. Des.*, vol. 35, pp. 157–168, 2000.
13. D. Merhof, R. Grosso, U. Tremel, and G. Greiner, Anisotropic Quadrilateral Mesh Generation: An Indirect Approach, *Adv. Eng. Software*, vol. 38, pp. 860–867, 2007.
14. P. L. Baehmann, S. L. Wittchen, M. S. Shephard, K. R. Grice, and M. A. Yerry, Robust Geometrically-Based, Automatic two-Dimensional Mesh Generation, *Int. J. Numer. Meth. Eng.*, vol. 24, pp. 1043–1078, 1987.
15. J. A. Talbert and A. R. Parkinson, Development of an Automatic Two Dimensional Finite Element Mesh Generator Using Quadrilateral Elements and Bezier Curve Boundary Definitions, *Int. J. Numer. Meth. Eng.*, vol. 29, pp. 1551–1567, 1991.
16. T. K. H. Tam and C. G. Armstrong, 2D Finite Element Mesh Generation by Medial Axis Subdivision, *Adv. Eng. Software*, vol. 13, pp. 313–324, 1991.
17. B. Joe, Quadrilateral Mesh Generation in Polygonal Regions, *Comput. Aid. Des.*, vol. 27, pp. 209–222, 1995.
18. T. Blacker, Automated Conformal Hexahedral Meshing Constraints, Challenges and Opportunities, *Eng. Comput.*, vol. 17, pp. 201–210, 2001.
19. Y. Zhang and C. Bajaj, Adaptive and Quality Quadrilateral/Hexahedral Meshing from Volumetric Data, *Comput. Meth. Appl. Mech. Eng.*, vol. 195, pp. 942–960, 2006.
20. S. J. Owen, M. L. Staten, S. A. Canann, and S. Saigal, Q-Morph: An Indirect Approach to Advancing Front Quad Meshing, *Int. J. Numer. Meth. Eng.*, vol. 44, pp. 1317–1340, 1999.
21. B. Cheng and B. H. V. Topping, Improved Adaptive Quadrilateral Mesh Generation Using Fission Elements, *Adv. Eng. Software*, vol. 29, pp. 733–744, 1998.
22. R. V. Garimella, M. J. Shashkov, and P. M. Knupp, Triangular and Quadrilateral Surface Mesh Quality Optimization Using Local Parametrization, *Comput. Meth. Appl. Mech. Eng.*, vol. 193, pp. 913–928, 2004.
23. X. M. Chen, S. Cen, Y. Q. Long, and Z. H. Yao, Membrane Elements Insensitive to Distortion Using the Quadrilateral Area Coordinate Method, *Comput. Struct.*, vol. 82, pp. 35–54, 2004.
24. C. Park, J. S. Noh, I. S. Jang, and J. M. Kang, A New Automated Scheme of Quadrilateral Mesh Generation for Randomly Distributed Line Constraints, *Comput. Aid. Des.*, vol. 39, pp. 258–267, 2007.
25. D. R. White and P. Kinney, Redesign of the Paving Algorithm: Robustness Enhancements through Element by Element Meshing, *Proc. 6th Int. Meshing Roundtable*, pp. 323–335, 1997.

26. S. A. Canann, J. R. Tristano, and M. L. Staten, An Approach to Combined Laplacian and Optimization-Based Smoothing for Triangular, Quadrilateral, and Quad-Dominant Meshes, *Proc. 7th Int. Meshing Roundtable*, 1998.
27. K. M. Dhanasekharan and J. L. Kokini, Design and Scaling of Wheat Dough Extrusion by Numerical Simulation of Flow and Heat Transfer, *Int. J. Food Eng.*, vol. 60, pp. 421–430, 2003.