Applied Mathematical Sciences, Vol. 8, 2014, no. 171, 8493 - 8516
HIKARI Ltd, www.m-hikari.com
http://dx.doi.org/10.12988/ams.2014.49777

An Insight into the Science of Unstructured Meshes in Computer Numerical Simulation

[1]Z. Abal Abas, [2]S. Salleh, [1]S. Ilias, [1]A. F. N. Abdul Rahman,
[1]A. S. Shibghatullah, [1]B. Mohd Aboobaider and [3]S. Sidek

[1]Optimisation, Modelling, Analysis, Simulation and Scheduling (OptiMASS)
Research Group, Faculty of Information Communication and Technology
Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal
Melaka, Malaysia

[2]UTM Centre for Industrial & Applied Mathematics (UTM-CIAM), Universiti
Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia

[3]Centre for Languages and Human Development, Universiti Teknikal Malaysia
Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

Abstract

Computer numerical simulation is a beneficial tool for studying various domains of knowledge. Among the steps in the whole process of numerical simulation is the generation of unstructured meshes. Since the unstructured meshes are usually generated using automatic software, the fundamental knowledge of the unstructured meshes is often neglected. This paper highlighted some useful insights into the unstructured meshes in numerical simulation for several application domains, such as the radiative heat transfer problem, ocean modelling and biomedical engineering. It also reviewed some fundamental concepts and frameworks for element generation in producing unstructured meshes, particularly the Delaunay triangulation and advancing front techniques.

Keywords: Unstructured mesh, modelling, numerical simulation, Delaunay and Advancing Front technique

1.0 Introduction

Numerical simulation is the process of performing a program or multiple programs within one or more computers in order to represent the physical phenomenon based on its appropriate theoretical model. The models are often represented in governing mathematical equations. The whole processes of numerical simulation have indeed assisted us in studying the system in terms of its ‘desired’ aspects, operations and properties which can lead to evolution prediction.

Once a better understanding of the system being studied is achieved through numerical simulation process, a relevant decision can be made. Nowadays, the availability of the technology advancement especially in computer processing efforts has made it available for us to conduct desired numerical simulation studies in almost all domains of knowledge and applications. Such application areas range from various engineering fields, human science and environmental monitoring.

In general, the stages for numerical simulation can be broken down into pre-processing, processing or solving and post-processing. The first stage, which is called the pre-processing, involves generating the computational domain. This is followed by mesh processing and finally selecting appropriate governing equation of physical phenomena that need to be modelled. All of the required properties of the system as well as the conditions of the boundary need to be specified. The second stage involves the integration of the governing equations over all the control volumes or elements. This is followed by the discretisation of the governing equations into algebraic equations and solving the algebraic equations using an iterative method. Among the discretisation techniques are finite elements, finite volumes, finite difference, level set methods and others. Finally, the post-processing stage involves the visualisation of the simulation results. As mentioned in the first stage of numerical simulation above, mesh generation is conducted at the pre-processing stage. Mesh generation is the process of decomposing the whole computational domain into smaller connected elements.

Figure 1 illustrates examples for mesh generation towards computational domain.

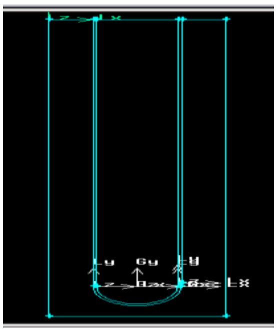


Fig. 1a: Computational domain.

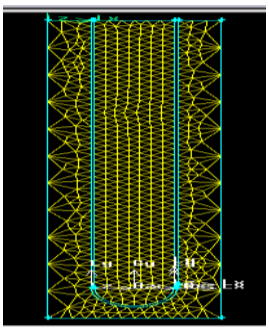


Fig. 1b: The generated mesh in the computational domain.

For every element of the meshes, the desired parameters which represent the state of the computational domain are defined. These elements are related to the parameters of the neighbouring elements by the respective algebraic equations. There are two types of mesh generation: structured and unstructured mesh generation.

Although the mesh generation is just one of the many activities in the pre-processing stages, it is a very crucial procedure since the accuracy of the numerical simulation and convergence rate depends on it. This paper aims to provide some useful insights into the science of unstructured mesh as well as its significance in certain numerical simulation applications. This paper is organised as follows: Section 2.0 provides selected examples on numerical simulation using unstructured meshes, while Section 3.0 describes in detail the two techniques for generating unstructured meshes: Delaunay and Advancing Front.

2.0 Examples of Numerical Simulation Application Using Unstructured Meshes

Numerical simulation has played a tremendous contribution towards various domains of knowledge. Among the domains that will be shared in this section are the radiative heat transfer, ocean modelling and biomedical engineering. These numerical simulations utilise mesh generation as a part of the simulation work.

2.1 Modelling in petrochemical industry

In chemical industry, numerical simulation has helped the engineers and researchers to study all kinds of chemical interaction processes, for example in the ethylene cracker furnace. Ethylene cracker furnace produces petrochemical products such as ethylene through a steam cracking process. They usually use computational fluid dynamics (CFD) to simulate the chemical reaction, heat flow, heat transfer and combustion processes in the furnace. In order to perform a computer numerical simulation in the ethylene furnace, its geometrical aspects of the furnace is transformed into a computational domain.

The unstructured mesh is then generated to discretise the computational domain of the furnace including the firebox and the reactor tubes. The resulted meshes are integrated with discretised Navier-Stokes equations in which the results of the simulation are often the detailed distribution of flow field, concentration field, heat flux and temperature field in the firebox as well as in the reactor tubes [1] – [11] The whole processes of numerical simulation which involve the unstructured mesh generation at the early stages can improve the furnace design for better performance as well as to assist the technical decision to increase product yield.

2.2 Ocean Modelling

Looking at the greater environment aspects, the hydrosphere of our world is made up of ocean, Shelf Sea, estuaries, rivers, groundwater and sea ice. Each of these components has its own governing equation in the form of partial differential equation which is derived from fluid mechanics. The study of these components has significant application in climate change study as well as environment impact assessment [12] – [15]. Unstructured mesh has become a preferred technique for domain representation and non-uniform resolution [16]. In detail, it can resolve simultaneous simulation of flow for both the small and large ocean scales in varying resolution and represent complex coastlines and bathymetry [17].

The work in [18] presented a complex tidal modelling in which the tidal equations are discretised over an unstructured triangular mesh and solved using a finite element. Much earlier work has been focusing on the tidal and costal problems with regard to shallow water models [12]. However, the work in [19] raised the issue of high computational cost for having unstructured meshes; therefore, they suggested an adaptive anisotropic mesh based on optimisation approach for ocean modelling. More recently, a global application of ocean general circulation model has been carried out in [15] with the use of unstructured meshes with finite element discretisation called the Finite Element Sea ice-Ocean Model (FESOM). The ocean modeling using unstructured meshes has been an ongoing research and there are many new models being announced respectively.

2.3 Biomedical Engineering

Combining biomedical engineering such as medical image analysis with clinical practice can be regarded as a multidisciplinary approach. Medical imaging analysis involves the process of acquiring medical image from medical imaging technology followed by performing numerical analysis to gain desirable medical information. Once the medical image has been acquired, a geometrical model which involves unstructured mesh generation has to be developed. As for biomedical engineering, numerical simulation can be seen as a supporting the planning of the operational procedure and predicting the outcomes. This is one of the significance of having medical image analysis and patient specific simulation.

Auricchio et al. [20] proves that this approach can enhance the success of endovascular treatment for ascending aortic pseudo aneurysms as well as for other aortic diseases by using finite element analysis. Another example for medical image analysis is the cardiac imaging where the cardiac tissue is discretised into finite element using mechanical laws of Hook's law [21]. An analysis focusing on a study of biomechanical behaviour of the spine, particularly the Intervertebral Disc (IVD) using Voxel as a meshing technique together with the use of finite element as its discretisation procedure is presented in Cortez et al. [22].

It can be noted that medical imaging analysis also plays a significant role in medical education, training and preoperative diagnostics. Djukic et al. [23] proposed a method utilising Virtual Reality (VR) technology to visualize the result of fluid flow using finite element analysis. They were able to visualize the shear stresses on the artery walls and perform simulation of motion through the artery. This shows that medical imaging analysis definitely can assist the experts to have a better perception of the current state of the subject matter, hence addressing any anomalies. An app has been developed by Fraunhofer MEVIS that utilises VR technology. It provides surgeons with a real-time and interactive access to the patient's data leading to a safer liver surgery [24]. It is anticipated that virtual reality will be among the next top methods in medicine. Its objective is to recreate the representation of reality to create a real image.

3.0 The Unstructured Meshes

As has been presented in the previous section, we can see that mesh is a crucial procedure in numerical simulation process. Structured or unstructured meshes are an important step for approximating solution to boundary value problems. In comparison of the two meshes, the unstructured meshes are becoming more predominant due to their ability to model complex geometry and adapt in natural environment [25], [26]. Furthermore, unstructured meshes are more preferable for tight industrial project schedule as they can be generated quickly [27]. Among the famous approaches for unstructured mesh generation are the Delaunay and Advancing Front Techniques.

Owen has conducted a broad study regarding mesh generation algorithm which is considered as a fundamental aspect of the field [28]. He also conducted a survey on the unstructured mesh generation software technology [29]. However, there has been not much work in reviewing advancing front technique. Although there are many approaches in generating unstructured meshes, this paper will only present Delaunay and Advancing Front in order to provide readers the fundamental concept of constructing the elements of unstructured meshes.

3.1 The Delaunay mesh generation with its improvement methods

The Delaunay triangulation is a method of generating elements by adding or inserting points in the interior of the computational domain and reconnecting them to form the element [25], [30] – [32]. The Delaunay triangulation has an important feature known as the Circumcircle Property or Delaunay criterion. For this type of triangulation, the vertices of a triangle cannot lie within the circumcircle of another triangle [25], [31], [33].

Examples of elements which satisfy and do not satisfy the circumcircle property or Delaunay criterion are illustrated in Fig. 2a and Fig. 2b. It must be noted that the Delaunay criterion is not an algorithm for mesh triangulation but it provides the criteria to connect the existing points in the domain [29]. With this feature, the minimum angle is maximized for all triangles [34], [35]. Among the approaches of the Delaunay triangulation are the Bowyer-Watson algorithm and the incremental point insertion strategy.

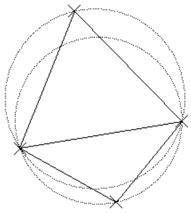


Fig. 2a: Elements that satisfy the circumcircle property.

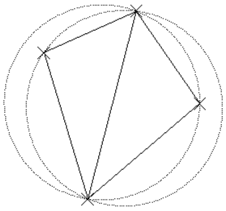


Fig. 2b: Elements that do not satisfy the circumcircle property.

The steps for Delaunay triangulation using the Bowyer-Watson algorithm are as follows [31], [34], [36], [37]:

1. Elements that contain the inserted point are located as in Fig. 3a.
2. All circumcircle elements that contain the new inserted point are deleted.
3. New elements are created by connecting the new inserted point to all points at the boundary of the convex cavity as in Fig. 3b.

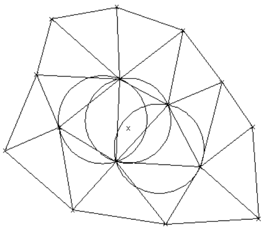


Fig. 3a: Detection of the elements which contain the new inserted point.

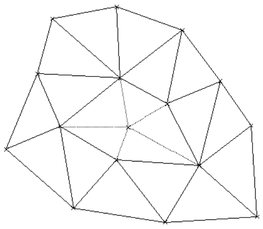


Fig. 3b: The creation of new elements by connecting the new point to the points at the boundary of the convex cavity.

Another well-known approach of Delaunay triangulation is the incremental point insertion strategy. Fig. 4 illustrates the steps for the incremental point insertion strategy and these steps are outlined as follows [34], [37]:

1. The boundary of the domain is discretised.
2. An initial Delaunay triangulation that covers all the boundary nodes is generated.
3. A list of triangles which violates the size or quality measurement is built, starting with the worst element.
4. A new point P is placed at the circumcentre of the first element in the list. The grid is then locally retriangulated. Each new element is checked and added in the list, if it does not belong to the range of acceptable quality or size.
5. Step 4 is repeated until there is no element in the list.
6. The element which is outside of the domain is deleted and the boundaries are recovered.
7. The mesh quality is checked. Smoothing or swapping is done, if necessary.

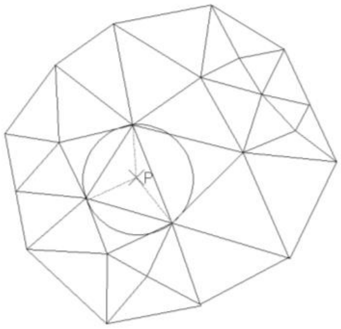


Fig. 4: The illustration for the incremental point insertion strategy.

A work done in [38] shows the algorithms for creating high quality triangular meshes on curved surfaces. There are two classifications for surface mesh generation algorithm: parametric space and direct 3 dimensions [29]. It can be seen that the two-dimensional Delaunay triangulation has reached the state of maturity and has been fully investigated. The three-dimensional Delaunay triangulation or the volume meshing is generated by extending the surface mesh algorithm. However, problems, such as the boundary recovery still persist when adopting the three dimensions [39]. A new algorithm for Delaunay refinement has been presented in [40] for brain image analysis where the algorithm manages to recover the object surface and meshes the interior volume simultaneously without sampling the object surface as the pre-processing step. Before the Delaunay triangulation is computed, the algorithm initially creates a box consisting of 8 vertices. The triangulation is the starting point of the refinement process, and during this process, some vertices are inserted exactly on the box. The box vertices and edges that are located precisely on the 12 edges of the box are called box edges. The box vertices are divided into two categories: the box-edge vertices and non-box-edge vertices. Meanwhile, the box facets are facets that lie precisely on 1 of the 6 facets of the box. There are also two tetrahedral defined as the intersecting tetrahedral and skinny tetrahedral. Tools such as Insight Toolkit (ITK), Computational Geometry Algorithm Library (CGAL), Visualization Toolkit (VTK) and Paraview were used to support the process. However, the elimination of slivers is still a problem that has yet to be resolved.

Recently, there are numerous works that utilise Delaunay triangulation. A flexible new mesh generation framework for image representation with two concrete methods named as the ID1 and ID2 have been proposed in [41]. By having varying parameters, the mesh quality can be increased or decreased according to an increased or a decreased computational cost. The framework highlighted several degrees of freedom which they leave them as an open choice for three criteria: initial mesh, growth schedule and candidate selection policy.

Another more recently work presented in [42] focused on the development of a parallel high quality Delaunay Image-to-Mesh Conversion (PI2M) Iso-surface based algorithm. The work initially started from a multi-labelled segmented image then simultaneously recovered the isosorface(s) ∂o of the object(s) o and meshes the volume. One of the characteristics that made this work stands out from the rest is that it does not perform mesh in the polyhedral domain, but rather performs both the volume and isosurface mesh directly from an image.

While the work of generating unstructured meshes is somehow has established, the work of improving the initial unstructured meshes or mesh optimisation methods is still a growing research area since the quality of the meshes affects the accuracy of simulation results. According to [43] the mesh quality improvement could be classified into three categories: (1) Topology optimisation (2) Vertex insertion or deletion (3) Vertex smoothing.

The original optimal Delaunay Triangulation is designed to improve the mesh quality by moving the inner vertices; hence no focus is dedicated at the boundary elements. To tackle this problem, the work in [44] proposed a new method that uses unified optimisation framework, named the Boundary Optimized Delaunay triangulation (B-ODT). The techniques integrate the vertex insertion operation with a minimum number of vertex. It computes the optimal boundary vertex by solving linear equation system for tetrahedral meshes. The same authors also illustrate a mesh smoothing algorithm by solving quadratic optimisation problem for feature preserving surface mesh, and it is named as the Suboptimal Delaunay Triangulation (S-ODT) [45].

There are various Delaunay-based mesh generation programs being developed for commercial and research work. Among the popular ones are the Triangle [46], TetGen [47], DistMesh [48], Gmsh [49], GRUMMP [50] and G23FM [51].

3.2 The Advancing Front Techniques and its Recent Approaches

3.2.1 The general basic steps of the standard Advancing Front Technique (AFT)

The first person to formulate the advancing front technique was George in 1971 [25], [30], [52], [53]. However, it was reported that this original publication failed to receive considerable immediate attention [52]. Lo in [54] extended this work by connecting a set of points which had been created beforehand. The algorithm by Lo was further modified by generating points and element simultaneously [25], [52]. The boundary curves are discretized by dividing the boundary curves into straight line segments or edges where nodes are placed on each of the segments or edges.

In this paper, an edge is denoted as (a,b) which means that the edge is formed by node a and b. Figure 5a illustrates a simple two-dimensional domain with connected boundary curves, while Figure 5b illustrates the discretized boundary curves into a set of edges denoted by $(1,2),(2,3),(3,4)$ until $(10,1)$.

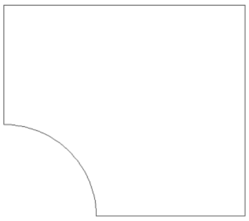


Fig. 5a: A two dimensional domain with connected boundary curves.

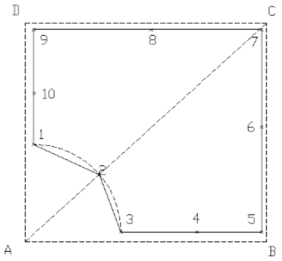


Fig 5b: A set of edges forming initial front and the background grid consists of triangle ABC and ADC.

A front [25], [52], [55], [56] is a list or a set of active edges which is currently available for element triangulation. With reference to Fig. 5b, all edges which are $(1,2),(2,3),(3,4)$ until $(10,1)$ form the first or initial front [52], [57]. The front is updated continuously when a triangular element is created. This can be seen in Fig. 6, taken from [58]. It must be noted that all edges belonging to the current front are considered as active edges while edges deleted from the current front list are considered as passive edges. Any nodes which compose the active edges in the current position of the front list are also known as active nodes.

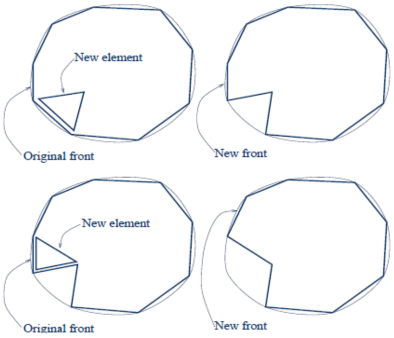


Fig. 6: The illustration of element with regards to front, taken from [58].

The main approach of controlling the grid in terms of the size and shape of triangular element involves the definition of the required grid-cell characteristic through background grid generation [25], [26], [30], [52], [56], [59]–[61]. The characteristics of the grid-cell, known as the grid cell parameters are the size parameter δ, the stretching s and the orientation of the cells ϕ [25], [52]. For example, Fig. 5b illustrates the background grid consisting of two triangles ABC and ADC. The grid cell parameters are required to be specified at each node forming the background grid, where in this example they are at nodes A, B, C and D. In the creation of the triangular element, once the edge is selected from the front for triangulation, the position of 'ideal point' (IP) [25], [52], [62], [63] on the perpendicular bisector of the edge is computed in such a way that an equilateral triangle is formed with IP as the third vertex. Fig. 7a illustrates the selected edge which is (a,b) and the position of the IP.

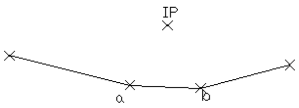


Fig 7a: The selected edge (a,b) and the position of IP.

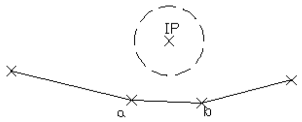


Fig. 7b. A circle is constructed with centre at IP and the radius of the circle following the empirical rule.

Grid cell parameters for all nodes in the front are interpolated [25]. The edge with the shortest length l in the front is selected as a departure zone in order to create a triangular element. In order to obtain δ for the next step of calculation, the interpolated values of δ corresponding to the two nodes of the edge are averaged. A circle with a centre at IP is constructed with the following empirical formula for the radius, $r = 0.8 \times \delta'$, where [25]

$$\delta' = \begin{cases} 0.55 \times l; & \delta < 0.55 \times l \\ \delta; & 0.55 \times l \le \delta \le 2.0 \times l \\ 2.0 \times l; & \delta > 2.0 \times l \end{cases} \tag{1}$$

Fig. 7b illustrates a circle with its centre at IP and the radius of the circle following the empirical rule. Active nodes that lie within the circle are searched (if any) and their distances from IP are listed. The best candidate for the third vertex of the triangular element would be the closest one to the IP. The equilateral triangle is constructed, if there are no such active nodes lying within the circle. However, the validity of the IP becoming the node must be checked by which it needs to satisfy the following conditions [25]: (1) The coordinate of IP does not lie inside another existing triangular element and (2) There is no intersection between the side of the new triangular element and any existing sides of the active front.

Generally, the front progresses into the interior of the domain in a ‘marching’ [25], [34], [52] process as a result of producing a triangular element in which new nodes and edges are created and at the same time old related edges are deleted. This process is illustrated in Fig. 8. A new triangular element is constructed by two nodes of a segment of a front and another node, either a newly created one or one that has already existed in the front. This process is continued until there are no longer any active edges in the front list. The general scheme of standard AFT is illustrated in Fig 9.

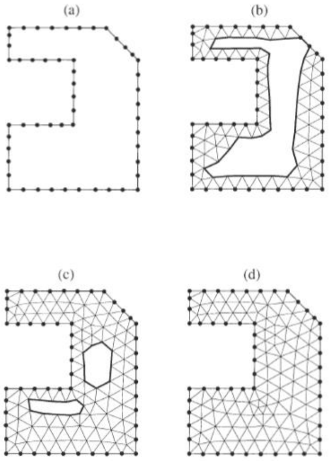


Fig 8: The illustration of the front 'marching' process into the interior of the domain, taken from [34].

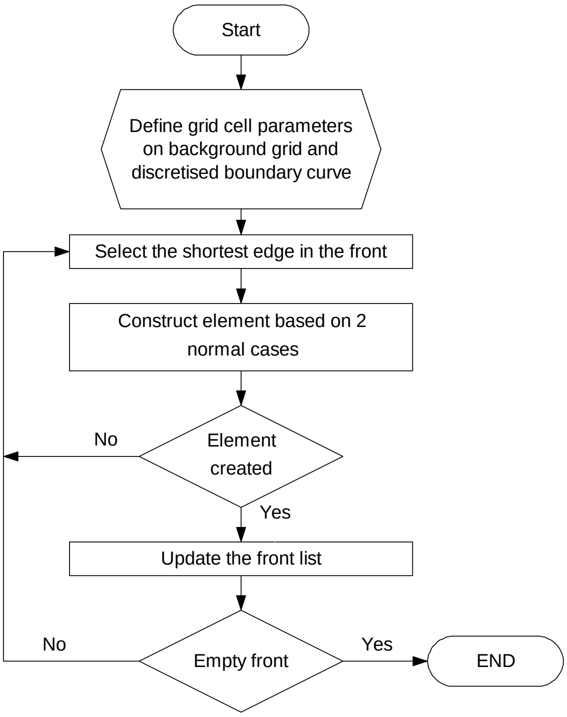

Start
Define grid cell parameters on background grid and discretised boundary curve
Select the shortest edge in the front
Construct element based on 2 normal cases
Element created
No
Yes
Update the front list
Empty front
No
Yes
END

Fig 9: The general scheme of standard AFT.

3.2.2 Recent approaches in the Advancing Front Technique

In [64], the triangulation process of a closed two-dimensional region was proposed. The method involves steps for generating the elements and reading their boundary values, smoothing the grids and labeling the nodes. The method focuses on the renumbering of the nodes to make way for the Cuthill-McKee algorithm[64] in estimating the matrix bandwidth [65], which are to be applied for producing smaller bandwidths in order to reduce the storage and amount of computation time.

Ito et al. [66] proposed a robust and low computational approach for a parallel framework to generate large-scale meshes for partitioning a coarse tetrahedral mesh into a number of sub-domains using a partitioning algorithm called METIS. The model is based on a parallel advancing front method to generate the volume mesh in each sub-domain.

Sazanov [67] proposed a new model which generates near boundary elements with their own modified form of advancing front techniques. This model allows the nodes to be located based on the length of the adjacent edges in its front and the angle between the edges. A complete row of elements are constructed with the placement of the nodes. Fig. 10 taken from [67] illustrates the determination of optimal number of the nodes to be generated around node n based on the length of adjacent edges $\delta_{n-1,n}$ and $\delta_{n,n+1}$ in the front and angle α_n between the edges. The optimal number of new points, N_n, is determined based on α_n and the complicated analysis procedure provide the ranged number of points. The N_n points are positioned on a perpendicular bisector of the two edges adjacent to node n. The distance $b_{n,n+1}$ of the potential point from the edge which joins node n and $n+1$ is determined based on an expression in [67]. The model also stitches together two different meshes to provide a consistent mesh for the whole domain.

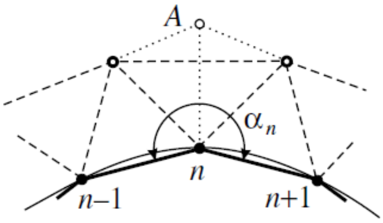


Fig. 10: The determination of optimal number of points taken from [67].

Another work that promotes the advancing front techniques but has a different approach from the conventional one is generating an umbrella of triangles centered at a point of a closed curve, called as Isotropic Umbrella Based Triangulation [68]. The ordered samples of points are constructed at the curve by using an implicit equation and following some of its rules. The new edges are formed by connecting the resulting selected points at the curve to the centre of the curve. This procedure is repeated until the entire domain is triangulated as can be seen in Fig. 11 taken from [68].

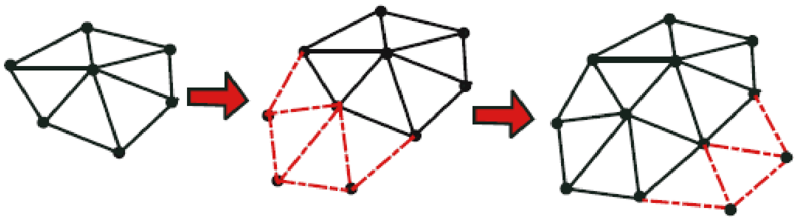


Fig. 11. The first local umbrella at the left and the extremely right is completing two local umbrella taken from [68].

If p is not the first center as can be seen in Fig. 12 in which that there are some vertices center at $F(p)$, a close segment needs to be constructed. A segment is called as a correct close segment if it does not contain enough vertex of the cell centered at that particular centre point. As in Fig. 12, $C(p)$ is considered as a correct segment between points p_s and p_e in order to close the two existing local umbrellas.

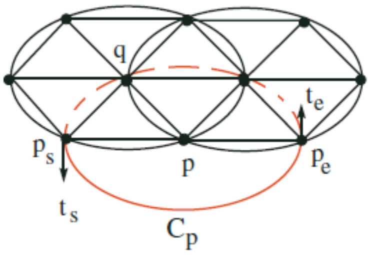
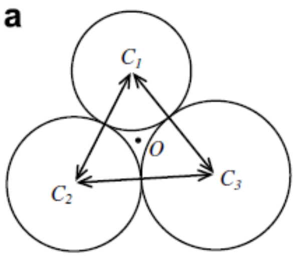


Fig. 12: Choosing the correct segment at point p taken from [68].

A circle packing algorithm which also utilizes the advancing front method generates an unstructured mesh by connecting the centre of packing circles with specified sizes [69]. The procedure starts from any convenient point within an open domain, and it does not need to start from the boundary as proposed by the conventional frontal method. As an illustration, Fig. 13a – Fig. 13b are taken from [69]. The method places the first three circles tangent to each other around the origin as shown in Fig. 13a. This forms the first element of the mesh and initial front. The second element is formed when a new circle is added as in Fig. 13b. An example of mesh generation by the circle packing algorithm with random size is illustrated in Fig. 14a and Fig. 14b, taken from [69].



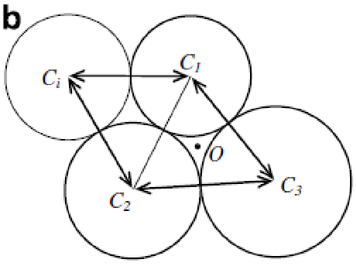
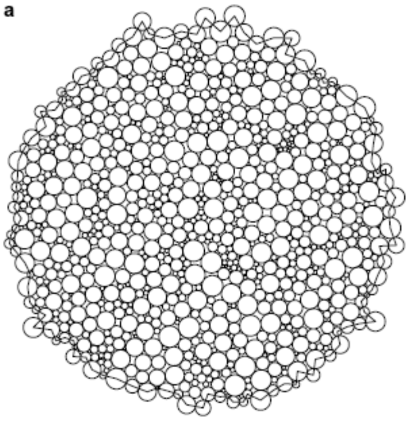


Fig. 13a: The first element is generated by connecting the centre of the circle and become the initial front, taken from [69].

Fig. 13b: The second element is generated by adding a new circle and connect the centre of the circle to the existing element, taken from [69].



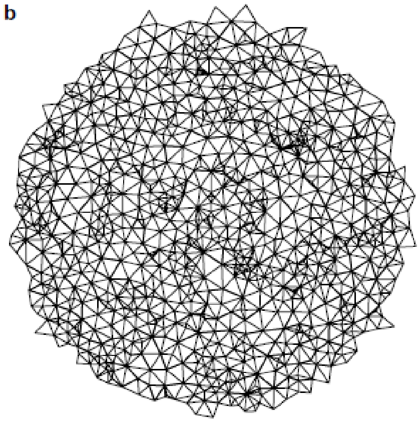


Figure 14a. The packing of circles with random size across the domain, taken from [69].

Figure 14b. The resulted unstructured mesh with random size, taken from [69].

Apart from the method of handling the construction of the triangular element, one of the aspects that needs attention when generating the unstructured mesh is the adequate grid control in terms of acceptable size and shape of the triangular element. A background mesh is reported to be among the most common methods of defining an element sizing function [25], [52], [59],[70],[71],[72]. It does not have to be aligned with the boundary of the domain as long as it satisfies the condition of being able to cover the domain to be discretized. It can be a coarse mesh, even for a complex computational domain. These background meshes usually consist of a small number of elements and can even consist of only one element.

Instead of specifying the value of cell parameter in the background mesh, the distribution of sources [25], [52], [70] at certain regions would be favorable. A spatial distribution of grid-cell size is specified as a function of distance from a point of interest to a ‘source’, in which the source can be a point, line segment or triangle. A background overlay grid size function is introduced in [59] where the defined size function generates a background overlay and then it is used as the base grid for determining the mesh size at all points during the process of meshing. To be specific, the mesh size at any given point is found by interpolating pre-determined sizes at the corner points of the background cell in which the points fall.

The concept of control space is described in [56], [60], [73] and it is a method for associating a step-size in the whole domain in which a function defining the desirable size is associated at each point. The size of elements created is adapted to the requirement of the defined control space. Therefore, the corresponding function determines the progression of the element in terms of its size and shape [56]. An improvement was made by [74], [75] where the function of the mesh size derived from the combination of the control space concept and the sphere concept is not assigned in every point. Instead it is assigned to only the most critical region of a given domain. Boundary segments are obtained from the intersection of generated spheres with region boundary. Additionally, the field point creations are generated according to the information encoded in the control space [74], [46].

The work presented in [76] shows an extension of classical advancing front techniques in terms of its data structures and algorithm to mesh composite surface in the context of meshing constraints topology. The adaptation of advancing front technique is based on these three rules: (1) elements are created across multiple parametric surfaces; (2) all entities of mesh are associated with their image in all components (reference entity) of the composite surface; and (3) intersection assessments between 3D segments are executed in the parametric domain of their images.

A technique of parallel mesh generation using advancing front method based on master/slave model is presented in [77]. The technique applies, firstly a coarse quadtree in order to decompose the domain, and then uses a serial of advancing front to generate the mesh in each domain simultaneously. The technique also uses shift-and-remesh procedure in which each sub-domain must be shifted to a Cartesian direction when advancing the front to a neighbouring sub-domain. The same advancing front method is done on the shifted sub-domain until mesh is generated in the whole domain. The author claims that the same approach can be extended to perform three-dimensional mesh generations.

There are also some conditions where the whole domain could not be meshed when using standard advancing front techniques, especially when it involves highly non-convex initial geometries that leave some disconnected nonmeshable cavities. The work done in [78] presented a repairing procedure after standard advancing front techniques have been applied. The procedure involves a number of heuristic operators, such as node-insertion operators, Laplacian smooting operator, widening operator, surface smoothing operator, peak-removal operator, alternative node-insertion operator, partial cavity discretization operator and final iterative repairing procedure.

Another work that generates unstructured meshes using the approach of advancing front techniques, called the Seven Cases Unstructured Triangulation Technique (7CUTT) is presented in [10]. The work produced seven structured cases as a rule to create a triangular element for initial meshes. The cases made used all conditions of the objects such as the existing of nodes, the length of the edges and the condition of intersected active edges in order to come out with the triangular element creation procedure.

Despite the ability of advancing front techniques to generate robust mesh element, the efficiency of handling massive elements in order to determine the validity of element created during the process still requires some improvement. [61] has proposed a dynamic approach of handling the massive element checking by creating a simple scheme of domain partitioning and store a variable number of objects at individual cell. The approach includes devising a dynamic marking and unmarking the cell which involves the line segment to be considered for element validity, hence the changing generation front during mesh generation can be coped. Therefore, the element checking for validation would be minimized.

4.0 Conclusion

Among the points that we would like to highlight here is that the study of geometry in the past [79]–[82] has become significant in today's computational analysis as it provides valuable insights in various application domains with the advancement of computer processing ability. The purpose of this paper is to share with readers the significance of unstructured meshes in numerical simulation study. On the other hand, it cannot be denied that besides its routine use in various engineering applications, the development of mesh generation has been well established. However, conducting automatic mesh generation without its fundamental theory knowledge is insufficient. Therefore, this paper offers some understandings of its theoretical knowledge for those who would like to conduct numerical simulation involving mesh generation. In this regard, the Delaunay approach and Advancing Front Techniques have been considered since most of the mesh generation automation is regarded as a black box.

Acknowledgements. The authors thank the Universiti Teknikal Malaysia Melaka (UTeM) and Ministry of Higher Education Institutions, Malaysia for sponsoring this research work under the research grant project FRGS/2/2013/ICT07/FTMK/02/7/F00190.

References

[1] H. Guihua, W. Honggang, and Q. Feng, “Numerical simulation on flow , combustion and heat transfer of ethylene cracking furnaces,” *Chem. Eng. Sci.*, vol. 66,
no. 8, pp. 1600–1611, 2011. http://dx.doi.org/10.1016/j.ces.2010.12.028

[2] Y. Han, R. Xiao, and M. Zhang, “Combustion and Pyrolysis Reactions in a Naphtha Cracking Furnace,” *Chem. Eng. Technol.*, no. 1, pp. 112–120, 2007. http://dx.doi.org/10.1002/ceat.200600191

[3] C. Li, T. Qiu, B. Chen, L. Tian, and B. Song, “Fluid dynamic numerical simulation coupled with heat transfer and reaction in the tubular reactor of industrial cracking furnaces,” *Int. J. Numer. Methods Fluids,* no. April 2009, pp. 355–373, 2010. http://dx.doi.org/10.1002/fld.2017

[4] W. Nouwen, E. Dupon, S. Barendregt, and F. Waterreus, “Simulation tools evaluate large-capacity furnace designs,” *Oil Gas J.*, 1998.

[5] J. Aminian, S. Shahhosseini, and M. Bayar, “Investigation of Temperaure and Flow Fields in an Alternative Design of Industrial Cracking Furnaces using CFD,” *Iran. J. Chem. Eng.*, vol. 7, no. 3, pp. 61–73, 2010.

[6] K. M. Van Geem, M. F. Reyniers, and G. B. Marin, “Challenges of Modeling Steam Cracking of Heavy Feedstocks,” *Oil Gas Sci. Technol.*, vol. 63, no. 1, pp. 79–94, 2008. http://dx.doi.org/10.2516/ogst:2007084

[7] G. Stefanidis, B. Merci, G. Heynderickx, and G. Marin, “Gray/nongray gas radiation modeling in steam cracker CFD calculations,” *AIChE J.*, vol. 53, no. 7, pp. 1658–1669, 2007. http://dx.doi.org/10.1002/aic.11186

[8] A. J. M. Oprins and G. J. Heynderickx, “Calculation of three-dimensional ow and pressure fields in cracking furnaces,” *Chem. Eng. Sci.*, vol. 58, pp. 4883–4893, 2003. http://dx.doi.org/10.1016/j.ces.2002.12.006

[9] A. Habibi, B. Merci, and G. J. Heynderickx, “Impact of radiation models in CFD simulations of steam cracking furnaces,” *Comput. Chem. Eng.*, vol. 31, pp. 1389–1406, 2007. http://dx.doi.org/10.1016/j.compchemeng.2006.11.009

[10] A. S. H. Basari, Z. A. Abas, and Shaharuddin Salleh, “Seven Cases Unstructured Triangulation Technique for Simplified Version of Conceptual Model of Ethylene Furnace for Radiative Heat Transfer Approximation,” *Int. J. Comput. Appl.*, vol. 44, 2012.

[11] Z. Abal Abas, S. Salleh, and Z. Manan, "Extended Advancing Front Technique for the Initial Triangular Mesh Construction on a Single Coil for Radiative Heat Transfer," *Arab. J. Sci. Eng.*, vol. 38, no. 9, pp. 2245–2262, 2013. http://dx.doi.org/10.1007/s13369-013-0556-7

[12] M. Piggott, C. Pain, and G. Gorman, "Unstructured adaptive meshes for ocean modeling," *Geophys. Monogr.*, pp. 383–408, 2008. http://dx.doi.org/10.1029/177gm22

[13] J. Slingo, K. Bates, N. Nikiforakis, M. Piggott, M. Roberts, L. Shaffrey, I. Stevens, P. L. Vidale, and H. Weller, "Developing the next-generation climate system models: challenges and achievements.," *Philos. Trans. A. Math. Phys. Eng. Sci.*, vol. 367, no. 1890, pp. 815–31, Mar. 2009. http://dx.doi.org/10.1098/rsta.2008.0207

[14] E. Deleersnijder, "Beyond Ocean Modelling : Multi-Scale / Physics Numerical Simulation of the Hydrosphere I . Numerics and Preliminary Applications Motivations (I) • Hydrosphere : oceans, shelf seas, estuaries, rivers, groundwater," 2011.

[15] Q. Wang, S. Danilov, D. Sidorenko, R. Timmermann, C. Wekerle, X. Wang, T. Jung, and J. Schröter, "The Finite Element Sea ice-Ocean Model (FESOM): formulation of an unstructured-mesh ocean general circulation model," *Geosci. Model Dev. Discuss.*, vol. 6, no. 3, pp. 3893–3976, Jul. 2013. http://dx.doi.org/10.5194/gmdd-6-3893-2013

[16] S. Danilov, Q. Wang, M. Losch, D. Sidorenko, and J. Schröter, "Modeling ocean circulation on unstructured meshes: comparison of two horizontal discretizations," *Ocean Dyn.*, vol. 58, no. 5–6, pp. 365–374, Aug. 2008. http://dx.doi.org/10.1007/s10236-008-0138-5

[17] C. C. Pain, M. D. Piggot, A. J. H. Goddard, F. Fang, G. J. Gorman, D. P. Marshall, M. D. Eaton, P. w. Power, and C.R.E. de Oliveira, "Three-dimensional unstructured mesh ocean modelling," *Ocean Model.*, vol. 10, no. 1–2, pp. 5–33, 2005. http://dx.doi.org/10.1016/j.ocemod.2004.07.005

[18] F. Lyard, F. Lefevre, T. Letellier, and O. Francis, "Modelling the global ocean tides: modern insights from FES2004," *Ocean Dyn.*, vol. 56, no. 5–6, pp. 394–415, Sep. 2006. http://dx.doi.org/10.1007/s10236-006-0086-x

[19] M. D. Piggott, P. E. Farrell, C. R. Wilson, G. J. Gorman, and C. C. Pain, "Anisotropic mesh adaptivity for multi-scale ocean modelling.," *Philos. Trans. A. Math. Phys. Eng. Sci.*, vol. 367, no. 1907, pp. 4591–611, Nov. 2009. http://dx.doi.org/10.1098/rsta.2009.0155

[20] F. Auricchio, M. Conti, S. Marconi, A. Reali, J. L. Tolenaar, and S. Trimarchi, “Patient-specific aortic endografting simulation : From diagnosis to prediction,” *Comput. Biol. Med.*, vol. 43, no. 4, pp. 386–394, 2013. http://dx.doi.org/10.1016/j.compbiomed.2013.01.006

[21] V. Tavakoli and A. a. Amini, “A survey of shaped-based registration and segmentation techniques for cardiac images,” *Comput. Vis. Image Underst.*, vol. 117, no. 9, pp. 966–989, Sep. 2013. http://dx.doi.org/10.1016/j.cviu.2012.11.017

[22] S. Cortez, J. Claro, and J. Alves, “3D reconstruction of a spinal motion segment from 2D medical images: Objective quantification of the geometric accuracy of the FE mesh generation procedure,” *... (ENBENG), 2013 IEEE 3rd ...*, no. February, pp. 1–6, 2013. http://dx.doi.org/10.1109/enbeng.2013.6518443

[23] T. Djukic, V. Mandic, and N. Filipovic, “Virtual reality aided visualization of fluid flow simulations with application in medical education and diagnostics.,” *Comput. Biol. Med.*, vol. 43, no. 12, pp. 2046–52, Dec. 2013. http://dx.doi.org/10.1016/j.compbiomed.2013.10.004

[24] “Snapshot: Augmented Organs, BBC Knowledge,” *BBC Worldwide*, pp. 12–13, Feb-2014.

[25] M. Farrashkhalvat and J. P. Miles, *Basic Structured Grid Generation with an Introduction to Unstructured Grid Gerenation*. Butterworth-Heinemann, 2003.

[26] P. R. M. Lyra and D. K. Carvalho, “A computational methodology for automatic two-dimensional anisotropic mesh generation and adaptation,” *J. Brazilian Soc. Mech. Sci. Eng.*, vol. 28, no. 4, pp. 399–412, Dec. 2006. http://dx.doi.org/10.1590/s1678-58782006000400004

[27] S. Koric, Q. Lu, and E. Guleryuz, “Evaluation of massively parallel linear sparse solvers on unstructured finite element meshes,” *Comput. Struct.*, vol. 141, pp. 19–25, Aug. 2014. http://dx.doi.org/10.1016/j.compstruc.2014.05.009

[28] S. Owen, “An Introduction to Mesh Generation Algorithm,” in *14th International Meshing Roundtable*, 2005.

[29] S. Owen, “A Survey of Unstructured Mesh Generation Technology.,” *IMR*, 1998.

[30] R. Lohner, “Progress in Grid Generation via the Advancing Front Technique,” *Eng. Comput.*, pp. 186–210, 1996. http://dx.doi.org/10.1007/bf01198734

[31] D. Mavriplis, “Unstructured Mesh Discretizations and Solvers for Computational Aerodynamics (Invited),” *18th AIAA Computational Fluid Dynamics Conference*. American Institute of Aeronautics and Astronautics, Reston, Virigina, 25-Jun-2007. http://dx.doi.org/10.2514/6.2007-3955

[32] S. H. Lo, “Delaunay triangulation of non-uniform point distributions by means of multi-grid insertion,” *Finite Elem. Anal. Des.*, vol. 63, pp. 8–22, Jan. 2013. http://dx.doi.org/10.1016/j.finel.2012.08.005

[33] W. Hong, T.-S. Chen, and J. Chen, “Reversible data hiding using Delaunay triangulation and selective embedment,” *Inf. Sci. (Ny).*, Apr. 2014. http://dx.doi.org/10.1016/j.ins.2014.03.030

[34] J. Blazek, “Principles of Grid Generation,” in *Computational Fluid Dynamics: Principle and Applications*, 2007, pp. 353–392.

[35] A. Márquez, A. Moreno-González, Á. Plaza, and J. P. Suárez, “There are simple and robust refinements (almost) as good as Delaunay,” *Math. Comput. Simul.*, Jun. 2012.

[36] C. D. Antonopoulos, F. Blagojevic, A. N. Chernikov, N. P. Chrisochoides, and D. S. Nikolopoulos, “Algorithm, software, and hardware optimizations for Delaunay mesh generation on simultaneous multithreaded architectures,” *J. Parallel Distrib. Comput.*, vol. 69, no. 7, pp. 601–612, Jul. 2009. http://dx.doi.org/10.1016/j.jpdc.2009.03.005

[37] S.-W. Cheng, T. K. Dey, and J. Shewchuk, *Delaunay Mesh Generation*. CRC Press, 2012.

[38] L. Paul, “Guaranteed-Quality Mesh Generation for Curved Surfaces *,” pp. 274–280, 1993. http://dx.doi.org/10.1145/160985.161150

[39] Y. Liu, S. H. Lo, Z.-Q. Guan, and H.-W. Zhang, “Boundary recovery for 3D Delaunay triangulation,” *Finite Elem. Anal. Des.*, vol. 84, pp. 32–43, Jul. 2014. http://dx.doi.org/10.1016/j.finel.2014.02.006

[40] P. Foteinos, “Guaranteed quality tetrahedral Delaunay meshing for medical images,” *Vor. Diagrams*, vol. m, pp. 215–223, Jun. 2010. http://dx.doi.org/10.1109/isvd.2010.15

[41] M. D. Adams, “A highly-effective incremental/decremental Delaunay mesh-generation strategy for image representation,” *Signal Processing*, vol. 93, no. 4, pp. 749–764, Apr. 2013. http://dx.doi.org/10.1016/j.sigpro.2012.09.017

[42] P. A. Foteinos and N. P. Chrisochoides, “High quality real-time Image-to-Mesh conversion for finite element simulations,” *J. Parallel Distrib. Comput.*, vol. 74, no. 2, pp. 2123–2140, Feb. 2014. http://dx.doi.org/10.1016/j.jpdc.2013.11.002

[43] J. Leng, Y. Zhang, and G. Xu, “A novel geometric flow approach for quality improvement of multi-component tetrahedral meshes,” *Comput. Des.*, vol. 45, no. 10, pp. 1182–1197, Oct. 2013. http://dx.doi.org/10.1016/j.cad.2013.05.004

[44] Z. Gao, Z. Yu, and M. Holst, “Quality Tetrahedral Mesh Smoothing via Boundary-Optimized Delaunay Triangulation.,” *Comput. Aided Geom. Des.*, vol. 29, no. 9, pp. 707–721, Dec. 2012. http://dx.doi.org/10.1016/j.cagd.2012.07.001

[45] Z. Gao, Z. Yu, and M. Holst, “Feature-preserving surface mesh smoothing via suboptimal Delaunay triangulation.,” *Graph. Models*, vol. 75, no. 1, pp. 23–38, Jan. 2013. http://dx.doi.org/10.1016/j.gmod.2012.10.007

[46] J. R. Shewchuk, “Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator,” in *Selected Papers from the Workshop on Applied Computational Geormetry, Towards Geometric Engineering*, 1996, pp. 203–222. http://dx.doi.org/10.1007/bfb0014497

[47] H. Si, “TeTGent: A Quality Tetrahedral Mesh Generator and 3D Delaunay Triangulator.” 2013.

[48] P.-O. Persson and G. Strang, “A Simple Mesh Generator in Matlab.” 2004. http://dx.doi.org/10.1137/s0036144503429121

[49] C. Geuzaine and Jean-Francois Remacle, “Gmsh: A Three Dimensional Finite Element Mesh Generator With Buit In Pre and Post Processing Facilities,” *Int. J. Numer. Methods Eng.*, 2009.

[50] Carl Olliver Gooch, “GRUMMP Version 0.5.0 User’s Guide.” 2010.

[51] H. Mustapha, “G23FM: a tool for meshing complex geological media,” *Comput. Geosci.*, vol. 15, no. 3, pp. 385–397, 2011. http://dx.doi.org/10.1007/s10596-010-9210-6

[52] J. Peraire, J. Peiro, and K. Morgan, “Advancing Front Grid Generation,” in *Handbook of Grid Generation*, CRC Press LLC, 1999. http://dx.doi.org/10.1201/9781420050349.ch17

[53] J. A. George, “Computer Implemenattion of the Finite Element Method by STAN-CS-71-208 February, 1971 Computer Science School of Humanities and Sciences Stanford University,” 1971.

[54] S. H. Lo, “A new mesh generation scheme for arbitrary planar domains,” *Int. J. Numer. Methods Eng.*, vol. 21, no. 8, pp. 1403–1426, Aug. 1985. http://dx.doi.org/10.1002/nme.1620210805

[55] V. D. Liseikin, “Grid Generation Methods,” *Methods*, 2010.

[56] N. Kovač, S. Gotovac, and D. Poljak, “A new front updating solution applied to some engineering problems,” *Arch. Comput. Methods Eng.*, vol. 9, no. 1, pp. 43–75, Mar. 2002. http://dx.doi.org/10.1007/bf02736232

[57] G. Foucault, J.-C. Cuillière, V. François, J.-C. Léon, and R. Maranzana, “Generalizing the advancing front method to composite surfaces in the context of meshing constraints topology,” *Comput. Des.*, vol. 45, no. 11, pp. 1408–1425, Nov. 2013. http://dx.doi.org/10.1016/j.cad.2013.05.009

[58] P.-O. Persson, “Lecture 1 Computational Mesh Generation,” 2008.

[59] J. Zhu, T. Blacker, and R. Smith, “Background overlay grid size functions,” in *Proceedings of the 11th international meshing roundtable*, 2002, pp. 65–74.

[60] E. Seveno, “Towards an adaptive advancing front method,” in *6th International Meshing Roundtable*, 1997, pp. 349–362.

[61] S. H. Lo, “Dynamic grid for mesh generation by the advancing front method,” *Comput. Struct.*, vol. 123, pp. 15–27, Jul. 2013. http://dx.doi.org/10.1016/j.compstruc.2013.04.004

[62] Z. A. Abas and S. Salleh, “Enhanced Advancing Front Technique with Extension Cases for Initial Triangular Mesh Generation,” in *Proceedings of the World Congress on Engineering 2011*, 2011.

[63] R. Löhner, “A 2nd Generation Parallel Advancing Front Grid Generator,” in *Proceedings of the 21st International Meshing Roundtable SE - 27*, X. Jiao and J.-C. Weill, Eds. Springer Berlin Heidelberg, 2013, pp. 457–474. http://dx.doi.org/10.1007/978-3-642-33573-0_27

[64] H. Naji, “An improved advancing front algorithm for triangulating arbitrary two dimensional regions.pdf,” in *THe 17th National Computer Conference*, 2004.

[65] M. Ke, J. Chen, and H. Jin, “Research of Using Dynamic Programming in the Nodes Encoding Optimization,” *Information Engineering and Computer Science, 2009. ICIECS 2009. International Conference on*. pp. 1–4, 2009. http://dx.doi.org/10.1109/iciecs.2009.5363278

[66] Y. Ito, A. M. Shih, A. K. Erukala, B. K. Soni, A. Chernikov, N. P. Chrisochoides, and K. Nakahashi, "Parallel unstructured mesh generation by an advancing front method," *Math. Comput. Simul.*, vol. 75, no. 5–6, pp. 200–209, Sep. 2007.

[67] I. Sazonov, D. Wang, O. Hassan, K. Morgan, and N. Weatherill, "A stitching method for the generation of unstructured meshes for use with co-volume solution techniques," *Comput. Methods Appl. Mech. Eng.*, vol. 195, no. 13–16, pp. 1826–1845, Feb. 2006. http://dx.doi.org/10.1016/j.cma.2005.05.037

[68] V. Hernández-Mederos, P. Ángel, and J. Estrada-Sarlabous, "Isotropic umbrella based triangulation of regular parametric surfaces," *Numer. Algorithms*, vol. 48, no. 1–3, pp. 29–47, 2008. http://dx.doi.org/10.1007/s11075-008-9188-5

[69] W. X. Wang, C. Y. Ming, and S. H. Lo, "Generation of triangular mesh with specified size by circle packing," *Adv. Eng. Softw.*, vol. 38, no. 2, pp. 133–142, Feb. 2007. http://dx.doi.org/10.1016/j.advengsoft.2006.04.006

[70] O. T.O, "An algorithm for parallel unstructured mesh generation and flow analysis," Massachusetts Institute of Technology, 1996.

[71] U. Tremel, F. Deister, O. Hassan, and N. P. Weatherill, "Parallel Generation of Unstructured Surface Grids," *Eng. with Comput.*, vol. 21, no. 1, pp. 36–46, Nov. 2005. http://dx.doi.org/10.1007/s00366-005-0311-0

[72] K. Sørli, "A Review of Computational Strategies for Complex Geometry and Physics," 2002.

[73] P. L. George and E. Seveno, "The advancing-front mesh generation method revisited," *Int. J. Numer. Methods Eng.*, vol. 37, no. 21, pp. 3605–3619, 1994. http://dx.doi.org/10.1002/nme.1620372103

[74] L. Vasiliauskiene and R. Baušys, "Intelligent initial finite element mesh generation for solutions of 2D Problems," *Informatica*, vol. 13, no. 2, pp. 239–250, 2002.

[75] R. Kacianauskas, E. Stupak, and S. Stupak, "Application of adaptive finite elements for solving elastic plastic problem of SENB specimen," *Mechanika*, vol. 1, no. 1, pp. 18–22, 2005.

[76] G. Foucault, J.-C. Cuillière, V. François, J.-C. Léon, and R. Maranzana, "Generalizing the advancing front method to composite surfaces in the context of meshing constraints topology," *Comput. Des.*, vol. 45, no. 11, pp. 1408–1425, Nov. 2013. http://dx.doi.org/10.1016/j.cad.2013.05.009

[77] M. O. Freitas, P. a. Wawrzynek, J. B. Cavalcante-Neto, C. a. Vidal, L. F. Martha, and A. R. Ingraffea, “A distributed-memory parallel technique for two-dimensional mesh generation for arbitrary domains,” *Adv. Eng. Softw.*, vol. 59, pp. 38–52, May 2013. http://dx.doi.org/10.1016/j.advengsoft.2013.03.005

[78] L. D. Adamoudis, G. Koini, and I. K. Nikolos, “Heuristic repairing operators for 3D tetrahedral mesh generation using the advancing-front technique,” *Adv. Eng. Softw.*, vol. 54, pp. 49–62, Dec. 2012.
http://dx.doi.org/10.1016/j.advengsoft.2012.08.007

[79] R. Fitzpatrick, *Euclid’s Elements of Geometry*. 1885, pp. 1883–1885.

[80] P. Davis, “The rise, fall, and possible transfiguration of triangle geometry: a mini-history,” *Am. Math. Mon.*, 1995. http://dx.doi.org/10.2307/2975007

[81] R. Sidik, M. Sidek, and J. A. Bakar, “Al-Khawarizmi: Leading Contributor To Indigenisation of Science in The Islamic Civilization.,” *... Nat. Appl. Sci.*, vol. 6, no. 3, pp. 324–330, 2012.

[82] A. Baki, “AL Khwarizmi’s Contributions to the Science of Mathematics: AL Kitabaljabrwa'l Muqabalah,” pp. 225–228, 1992.

Received: September 15, 2014, Published: November 28, 2014