

A new conforming mesh generator for three-dimensional discrete fracture networks

Soheil Mohajerani ^a, Seyed Mohammad Esmaeil jalali ^{a,*}, Seyed Rahman Torabi ^a

^a Department of Mining Engineering, Petroleum and Geophysics, Shahrood University of Technology, Shahrood, Iran

Article History:

Received: 03 January 2018,

Revised: 09 July 2018

Accepted: 14 July 2018.

ABSTRACT

Nowadays, numerical modelings play a key role in analyzing hydraulic problems in fractured rock media. The discrete fracture network model is one of the most used numerical models to simulate the geometrical structure of a rock-mass. In such media, discontinuities are considered as discrete paths for fluid flow through the rock-mass while its matrix is assumed impermeable. There are two main parameters to simulate the geometry of discontinuities in this model; the density and connectivity pattern of fractures. Despite the advantages of the discrete fracture network modeling, in order to apply the numerical solution schemes, the discretization of this model has encountered serious challenges due to the geometrical complexities. Generally, some of previous meshing methods present a framework that changes the geometrical structure and connectivity pattern of the model, and some others are incapable to mesh intricate networks with a large number of fractures. In this research, a new algorithm is developed to mesh the geometrical framework of three-dimensional discrete fracture networks. This algorithm is designed based on a refined conforming Delaunay triangulation and is computationally efficient, fast and low-cost. Furthermore, it never changes the geometrical structure of a DFN primitive model, therefore, the connectivity pattern will remain intact and the mesh is a proper representative of DFN. The algorithm was validated using the analytical methods and a series of sensitivity analysis was performed to evaluate the effect of meshing parameters on fluid flow using a finite element scheme of steady state Darcy flow. The results show that an optimized minimum internal angle of meshing elements should be predetermined to guarantee termination of the algorithm.

Keywords : Meshing, Triangulation, Conforming mesh, DFN, Delaunay, FEM

1. Introduction

Numerical simulations of fluid flow through fractured rocks play a vital role in many applications in the energy industry, such as hydrocarbon reservoirs, geothermal resources, underground fluid storage, groundwater aquifers, nuclear waste disposal and clearance of contaminated areas in the fractured rocks [1-7]. Generally, numerical models of the fluid flow through fractured rocks can be classified into three subcategories: the equivalent continuum model, the double medium model and the discrete fracture network (DFN). DFN is one of the most accurate and mostly used methods to simulate the fluid flow through fractured rocks [8-10]. In this method, the effect of discrete discontinuities on the fluid flow is explicitly considered with the impermeability assumption of rock matrix. DFN method is very popular because of its many advantages so that several models including a large number of fractures have been carefully studied using this method [11, 12].

The foundation of a DFN method is partitioning an n -dimensional domain to an $n-1$ -dimensional statistically distributed set of fractures. This partition has a significant effect on computational cost of the flow models, particularly for three-dimensional models. The fracture locations are generated in a desired domain using a statistical process, and the geometrical parameters of the fractures, such as, the orientation (dip and dip direction), and length are modeled using the probability density functions (PDF) based on sampling methods [7, 13]. Geological

datasets are directly obtained from boreholes, surface outcrops, trenches using one of the mapping methods (scanline, mapping window and circular estimator), and geophysical mappings are the most important part of a DFN modeling [14-16]. The shape of fractures is a hypothetical parameter and often simulated by circles, ellipses and polygons in various research works [17-19]. Hydrological parameters such as aperture and roughness of fracture wall surfaces may be estimated either by laboratory tests or using in-situ field tests. These parameters can be assigned to the location of fractures as a constant value or a PDF [20, 21].

In the literatures, various numerical solution schemes have been used to solve fluid flow problems in fractured rocks. In this regard, the finite element method (FEM) [11, 22], finite volume method (FVM) [23, 24], boundary element method (BEM) [25, 26], finite difference method (FDM) [27], discontinuous deformation analysis (DDA) [28,29] and hybrid methods [30-32] have obtained more attention. In general, such methods require a high quality meshing framework to solve the flow with an adequate precision.

Unfortunately, there are some serious challenges to discretize a DFN model into a high quality mesh. On one hand, the structured meshing is not convenient to represent a complex three-dimensional geometry of a fractured medium and on the other hand, a high quality unstructured meshing must be able to meet particular geometrical requirements [1, 33]. As shown in Fig. 1, a network of statistically generated fractures can include fractures length spanning over several orders of magnitude. In order to solve the flow field in small fractures, they must have small

* Corresponding author. E-mail address: jalalime@shahroodut.ac.ir (S.M.E. Jalali).

enough meshing elements compatible with their length. A good mesh requires a balance between two concepts: to avoid increasing the computational costs, its elements should not be too small or too large whereby the precision of the problem decreases. In addition, due to the complex structure of three-dimensional DFNs with arbitrary shape and spatial position of fractures, it is possible to form the intersection lines of fractures planes which are parallel or crossover on a third fracture plane. If the distance between parallel intersections or the angle between the crossover intersections is too small, a low-quality meshing may be generated. The meshing elements may lead to an ill-conditioned discretization matrix and cause divergence in numerical solution [34].

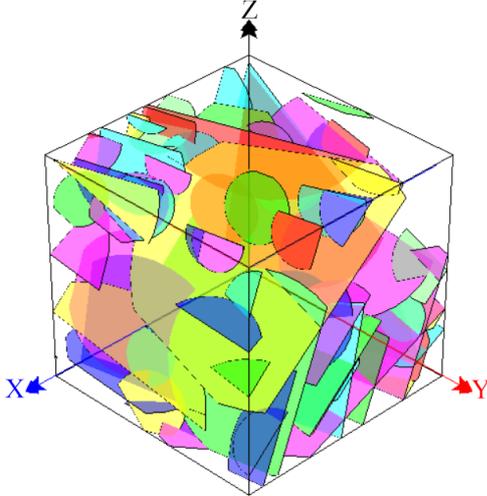


Fig. 1. A realization of DFN generated by the developed code in the research.

So far, many methods have been developed to mesh a DFN model. These methods are divided into two main subcategories, conforming and non-conforming methods. The conforming mesh refers to the case that nodes on the intersection line are unique and common to these two intersecting fractures. Instead, the non-conforming mesh discretizes each fracture plane independently. Although the non-conforming mesh is more flexible, an additional system of equations has to be implemented to ensure continuity of the hydraulic head and flow rate on the intersection of fractures; therefore, the solution scheme may be time-consuming [34,35].

Koudina et al. [24] provided one example of a conforming meshing algorithm for a DFN. An alternative algorithm based on the paving method was proposed by Wang et al. [8]. Although these methods are used successfully in simple networks of fractures, they cannot overcome the aforementioned challenges when the number of fractures increases and the geometrical structure of the DFN becomes more complex. The challenge was partially address by Maryška et al. [36]. However, in their method, the geometrical structure of the DFN changed and the intersections of fractures were changed with the length variation and displacement. It could change the connectivity pattern of fractures, therefore, it seems that the mesh was no longer a representative of the geometrical structure of the DFN.

Two similar meshing methods were also suggested by Mustapha and Mustapha [37] and Erhel et al. [38] in which the boundaries of fractures and intersection of them were first referred to regular cubes with constant mutual edges between the intersected fractures. Then, the cube that included the elements of a fracture was projected on the same fracture again. Although these methods could generate a high quality mesh, they were unable to model the intersection of more than two fractures. A generalization of these models was provided by Mustapha et al. in which the vertices of triangles in a two-dimensional space [33] or the tetrahedrons in a three-dimensional space [39] were displaced and merged inside regular cubes and were projected back on the fracture surfaces to improve the meshing quality. An alternative generalization of these methods was developed by Karimi-fard and Durlöfsky [40] in which a conforming mesh was generated using add, displacement, remove and merging of vertices of meshing triangles.

Another method was proposed by Hyman et al. [34] in which using the feature rejection algorithm (FRAM), the creation of inconvenient fractures was prevented during generating the DFN. However, the meshing challenges occurred in that method, and the geometry of the network completely changed; therefore, the connectivity pattern of fractures altered as well.

Li et al. [41] provided a method to generate a conforming mesh for the DFN in which using the Persson and Strang meshing generator, the location of vertices of triangles were determined through solving a system of equations of force balance in truss and resulted in a high quality mesh. This method was not optimal due to high computational load and was unable to completely overcome the meshing challenges.

Some studies have focused on developing the non-conforming meshing methods. As mentioned before, these methods have more computational costs than the conforming methods. Therefore, they are not generally appropriate for networks with a huge number of fractures [35, 42, 43]. Benedetto et al. [44] provided a combined conforming and non-conforming method to solve a fluid flow problem in the DFN using the virtual element method (VEM). In their method, some additional unique vertices were added on intersections of fractures and each fracture was meshed independently. The application of that method was limited to VEM.

The main purpose of this research is developing a new computationally efficient algorithm to mesh three-dimensional DFN structures. This algorithm never changes the geometrical structure of the network and thereby the connectivity pattern of fractures will remain unchanged. Accordingly, the meshing framework can be well representative of the original geometry of the DFN. Triangular elements of the meshing structure are generated based on the Delaunay criterion and are refined to increase the quality. Thus, discretizing the matrices assembled by the numerical schemes are not ill-conditioned and in most cases, the solution converges using this algorithm. Because of a huge number of fractures with a wide range of lengths in the DFN, this method is designed so that the size of triangles is neither smaller nor larger than an optimized range. Therefore, in addition to low computational costs, this method can achieve high numerical precisions. Furthermore, this algorithm is able to properly cover the critical meshing conditions such as conjunction of two fracture intersections with a small angle or two parallel fracture intersections with a small distance on the third fracture plane.

This paper is organized as follows: section 2 provides the algorithms of generating a DFN, finding the intersections of the fractures, and triangulation and refinement algorithms. Section 3 discusses on the details of the validation of the present algorithm with describing the finite element scheme, and elaborates how to solve the flow problem in two simple regular geometrical structures and compares the results with those of analytical methods. In addition, a series of sensitivity analyses on the meshing parameters in a complex DFN will be conducted to determine the effect of various parameters and to demonstrate the performance of the algorithm.

2. Methodology

In this section, various algorithm and methods used to develop the present meshing algorithm are separately described.

2.1. Generation of DFN

Depending on their geological origin, the rock fractures are grouped into joint sets that have similar geometrical properties (dip and dip direction). In a three-dimensional geometrical model, the joint sets were estimated using the hemispherical projection. The joint sets were simulated independently and the ultimate model was a union set of all of them. Each joint set included certain geometrical distribution parameters such as, the location, orientation (dip and dip direction) and length of planar fractures.

The location of fractures is the first parameter that must be considered in the simulation of joint-sets. A single-point homogeneous

Poisson process is generally used to determine the location of fractures in the domain of model. Given a constant number as fracture intensity (λ) (the number of fracture planes per unit volume of the model), the location is distributed in a three-dimensional space. The average of Poisson distribution is calculated from Eq. 1 [7].

$$\mu = \lambda \cdot V_m \quad (1)$$

where, V_m is volume of the model. Since, the center of some of fractures are outside of the model domain, while their length is large enough to enter it and affect the connectivity pattern, the generation domain of the fractures is initially considered several times larger than the model domain. The cube model is extracted from the generating domain after the completion of generating process. In order to generate the location of fractures, a random variable (η) from Poisson distribution function is generated using Eq. 2 [7].

$$P(\eta = n) = e^{-\mu} \frac{\mu^n}{n!} \quad (2)$$

A sequence of random numbers (x_i) in the range of [0,1] as far as, $\prod_{i=1}^k x_i < e^{-\mu}$, are generated using a uniform distribution function. For any $n = k$ events, three independent values are calculated by setting P in the uniform distribution function. These values are considered as coordinates of the location of the fracture (o). Fig. 2 demonstrates an image of the generated locations of DFN fractures.

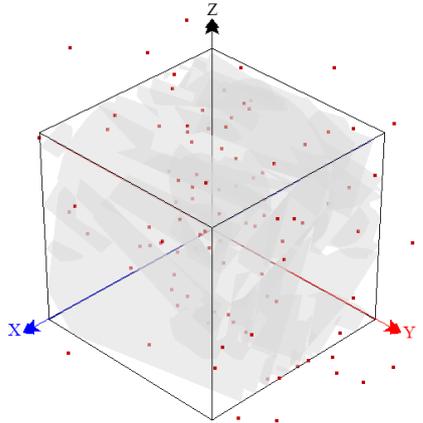


Fig. 2. Locations of DFN fractures generated using developed code in this research.

After that, the orientation and the length of fractures are generated using PDF and Monte-Carlo sampling, and are assigned to the locations of the fractures. The parameters, dip (α), dip direction (β) and rotation angle (γ) have been schematically shown in Fig. 3. The uniform and Fisher PDF are usually used to model the dip and dip direction respectively. The rotation angle is modeled by a uniform PDF as well.

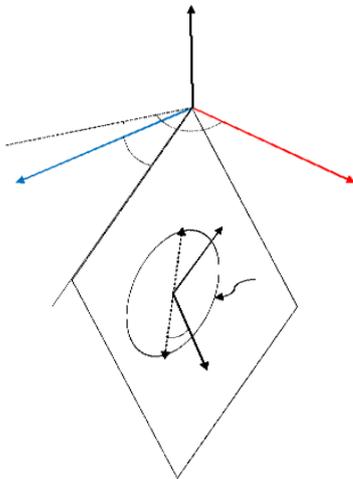


Fig. 3. A schematic fracture, its geometrical parameters and the global (X,Y,Z) and local (x,y) coordinate systems.

In the literatures, the power-law or log-normal is used as PDF of the fracture length (L). The shape of fractures is a hypothetical parameter which is simulated as circular, elliptical or polygonal shapes (Fig. 3). Then, the hydraulic parameters of fractures such as, aperture and roughness is assigned to the location of fractures as required similar to the generation of geometrical parameters. The aperture PDF is usually uniform [45]. Several equations have been suggested to determine the relationship between the aperture and the length of fractures. An example of such equations has been represented in Eq. 3 [46].

$$a = \zeta \sqrt{L} \quad (3)$$

Where, a is the aperture and ζ is a constant coefficient which is determined depending to the conditions of fractures and in general cases is equal to 0.004. Various PDFs and their parameters that are required to generate DFN have been listed in Table 1.

Table 1. PDF and their parameters.

PDF	Formula	Parameters
Uniform	$f(x) = \begin{cases} \frac{1}{b-a} & a \leq x \leq b \\ 0 & \text{otherwise} \end{cases}$	a, b
Fisher	$f(\theta) = k \sin \theta e^{k \cos \theta} / e^k - e^{-k}$	$\theta_{average}, k$
Normal	$f(x) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} & x > a \\ 0 & x \leq a \end{cases}$	a, μ, σ
Log-normal	$f(x) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma e^x} e^{-\frac{(\ln(x-a)-\mu)^2}{2\sigma^2}} & x > a \\ 0 & x \leq a \end{cases}$	a, μ, σ
Power-law	$f(x) = ax^{-k}$	a, k
Negative Exponential	$f(x) = \begin{cases} \lambda e^{-\lambda(x-\mu)} & x > \mu \\ 0 & x \leq \mu \end{cases}$	λ, μ

As mentioned before, after completing the process of generating the fractures, the original model domain is extracted from the primary generated domain. Each fracture and the set of all the fractures of the model are shown by f_i and $F = \bigcup_{i=1}^{N_f} f_i$, respectively, in which N_f is the number of all fractures. Regarding Fig. 3, two coordinate systems of are introduced in this research. Earlier is a global Cartesian system (X, Y, Z) defined in a three-dimensional space and later is locally defined on each fracture plane in a two-dimensional space (x, y).

2.2. Determination of intersection of fractures

In this research, the fractures are considered planar and the intersection of two fractures is a linear segment. In order to determine the coordinates of two ends of an intersection segment in (X, Y, Z), each fracture is intersected by other fractures and the boundary facets of the model. Fig. 4-a shows how the intersections are formed. Each of intersections and the set of all of them are represented by $s_i = f_j \cap f_k$. $j, k = 1 \dots N_f$ and $S = \bigcup_{i=1}^{N_s} s_i$, respectively where, N_s is the number of all intersections of the model. Therefore, the domain is characterized as $F \cup S$.

Fractures can have one of three main types of the connectivity with other fractures or boundaries of the model: multiple connectivity (persistent fractures), only one connection (dead-end fractures), and no connection (single fractures). As Fig. 4-b shows, the persistent fractures (blue colored) usually have a larger length and several (at least two) intersections with the other entities. Such fractures can be intersected by the boundaries of the model or be connected to dead-end fractures and be completely located inside the model. However, dead-end (green colored) and single (red colored) fractures can have important effects on the ultimate strength and mechanical properties of the rock-mass; they do not have any significant effect on hydraulic properties [47]. Since hydraulic analyses are the main aim of generating the three-dimensional DFNs in this research, it is convenient to remove dead-end

and single fractures from the model domain. It dramatically increases performance and speed of solution, particularly if the model is dealt with a huge number of fractures. F is investigated to find the isolated and dead-end fractures. The fractures with these conditions and their corresponding s are removed from F and S respectively. This process continues until there is no fracture with such conditions. Fig. 4-c shows an image of found intersections in the model domain.

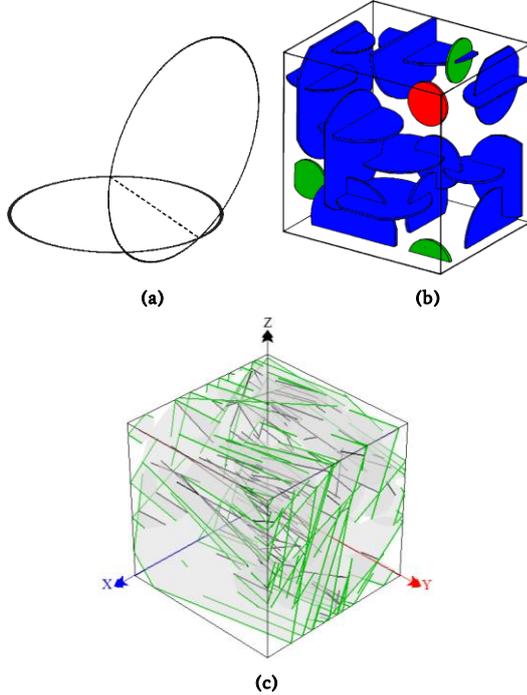


Fig. 4. Schematics of (a) formation of intersections, (b) isolated (red), dead-end (green) and persistent fractures [29], and (c) intersections of DFN fractures on boundaries of the model (green) and inside the model (black) found by the developed code in the research.

2.3. Mesh generation

The present meshing algorithm generates triangular elements in a two-dimensional (x, y) coordinate system on the surface of each fracture using the Delaunay criterion. The ultimate meshing geometry in the three-dimensional global (X, Y, Z) coordinate system is a union of all planar triangles which transpose using a transpose matrix. Since the vertices of the triangles on each intersection are unique, it results in a conforming mesh. To increase the quality of the triangulation, the Ruppert algorithm is used through refining low-quality triangles [48]. Elements of an unstructured triangulation do not follow a uniform pattern. On one hand, due to the complexity of the DFN geometry as well as the statistical shape and position of fractures and intersections in a three-dimensional space, and on the other hand, because of the tendency to not change the geometrical structure of the DFN during the mesh generation (to avoid changes in the fractures connectivity pattern), this algorithm provides an optimized unstructured triangulation.

The present algorithm includes four main steps:

- Step I; intersection vertices ($v_{s_i}^k$) are formed in S .
- Step II; Boundary vertices ($v_{b_i}^j$) are formed on the boundaries of the fractures (f_i).
- Step III; a Delaunay-based triangulation (T_i) is generated using a set of all vertices of fracture f_i , $V_i^j = (\cup_{j=1}^{N_{vs_i}} v_{s_i}^j) \cup (\cup_{j=1}^{N_{vb_i}} v_{b_i}^j)$, where N_{vs_i} and N_{vb_i} are the total number of intersection and boundary vertices of f_i .
- Step IV; T_i is refined using Ruppert algorithm.
- Steps I to IV are repeated for $i = 1 \dots N_f$.

According to the flowchart of Fig. 5, in the first step, s_i from the S is

intersected by the other intersections to find a possible crossover. If such a crossover is found, the vertex $v_{s_i}^k = s_i \cap s_j$, $i, j = 1 \dots N_s$ is registered and a coverage radius (h_s) is dedicated to it. h_s governs the mesh size and is determined arbitrarily by the user. More discussions are given regarding the determination of h_s in the following sections. Then, the middle point of the intersection segment s_i is identified using the coordinates of its two ends. If this point is not inside a sphere with a center and radius of $v_{s_i}^k$ and h_s respectively, a new vertex (\hat{v}_{s_i}) is registered with the coordinates of the center of this point and h_s is dedicated to it. In order to maintain the connectivity pattern of fractures, at least one of the vertices $v_{s_i}^k$ and \hat{v}_{s_i} must be saved if both of them overlap with previous registered vertices. After that, an iteration loop is created and for each iteration l , two vertices with spacing $l \times h_s$ are characterized on both sides \hat{v}_{s_i} and on the intersection s_i . These vertices are registered with the proviso that they are not inside a sphere with radius of h_s and center of previously saved vertices. These vertices are named $\check{v}_{s_i}^k$ and a coverage radius h_s is dedicated to them. The failure of the loop comes when the distance between two characterized vertices is larger than length of segment s_i . This process goes on until all unique intersection vertices in (X, Y, Z) are determined for all s_i , $i = 1 \dots N_s$. The set of these vertices is represented by $V_s = \cup_{i=1}^{N_s} (\cup_k v_{s_i}^k \cup \check{v}_{s_i}^k) \cup \hat{v}_{s_i}$.

In the second part, the algorithm is focused on f_i , $i = 1 \dots N_f$ from F . All intersections (s_j) of S included in f_i , are identified. Then, the vertices of s_j are selected from V_s and placed in V_i^j . The coordinates of these vertices is transposed from (X, Y, Z) to (x, y) on the plane of the fracture using a transpose matrix. After that, the vertices $v_{b_i}^j$ on the fracture boundaries (Γ_{f_i}) are characterized with the spacing h_s . These vertices are added to V_i^j and a h_s is dedicated to them if they are not inside a circle with the center and coverage the radius of previously registered vertices in V_i^j . At the end of this step, V_i^j is obtained with the total number of the vertices (N_{v_i}) on the fracture f_i .

A Delaunay-based triangulation can be generated with any arbitrary set of vertices [48]. Therefore, a Delaunay-based triangulation (T_i) is generated with V_i^j in the third step. In a two-dimensional space, T_i is Delaunay-based if and only if empty-circle criterion is satisfied for all elements of T_i [39]. This criterion checks whether the circumcircle of the j^{th} triangle (t_i^j) includes another vertex excludes t_i^j or not. Fig. 6 displays a schematic plan of the empty-circle criterion and the Delaunay-based triangulation. As shown in this fig., no vertex is inside the circumcircle of triangles. Moreover, three independent vertices of V_i form a triangle, must have the visibility property, that is, these vertices must be on the same open surface of s_i [48]. The visibility property causes independent meshing on each side of the intersections.

The basis of these refinement algorithms is to preserve triangulation as Delaunay by adding some vertices to reach a high quality triangulation. The Ruppert algorithm is one of the first refinement algorithms to improve the quality of a two-dimensional triangulation which is theoretically validated and practically satisfying. This algorithm finds low-quality t_i^j values and removes them from T_i during a forward searching process, and inserts a vertex in the circumcenter of removed t_i^j . Then, the searching process continues to remove triangles that lose their Delaunay property due to inserting the new vertex. Finally, a new triangulation (T_i^j) is generated with the new vertices whose corresponding triangle has been removed and T_i^j is added to T_i [49]. One of techniques to determine the quality of triangles is the ratio of the smallest edge to the radius of triangle's circumcircle (ω). Therefore, the minimum internal angle of the triangle can be calculated using Eq. 3 [48].

$$\theta_{min} = \arcsin\left(\frac{1}{2\omega}\right) \quad (3)$$

To ensure the termination of the Ruppert algorithm, the critical value of θ_{min} was theoretically calculated, 20.7° . If θ_{min} for each triangle is smaller than 20.7° , the triangle should be refined by the Ruppert algorithm. Refined triangulations for a single fracture with two orthogonal intersections, and a configuration of three orthogonal intersecting fractures are illustrated in Fig. 7.

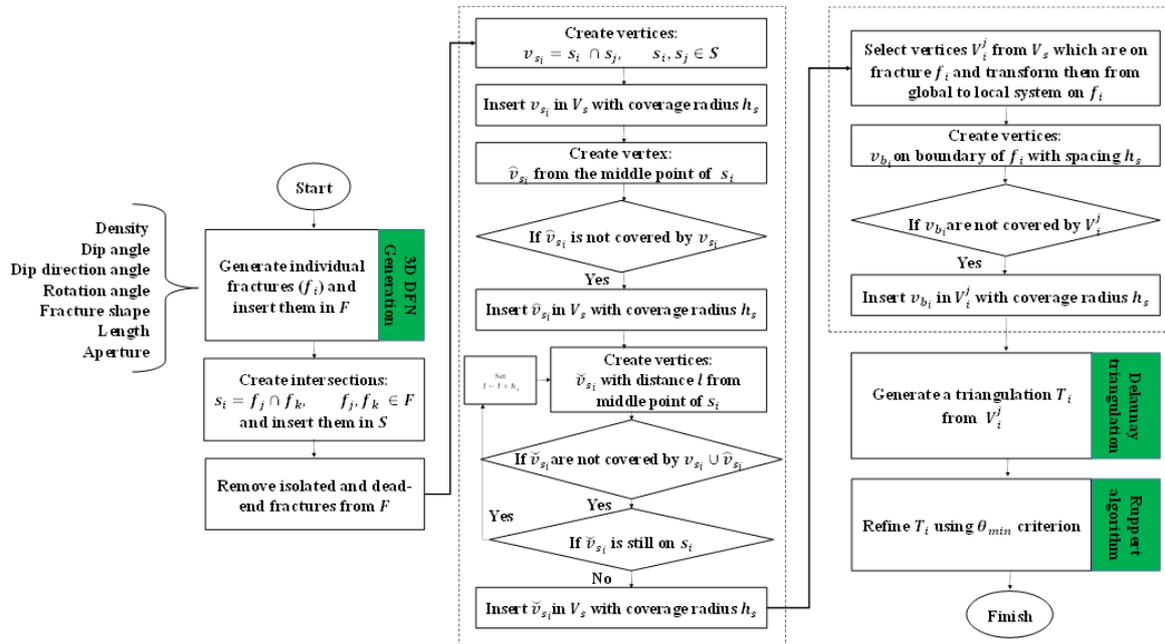


Fig. 5. Flowchart of the presented meshing algorithm.

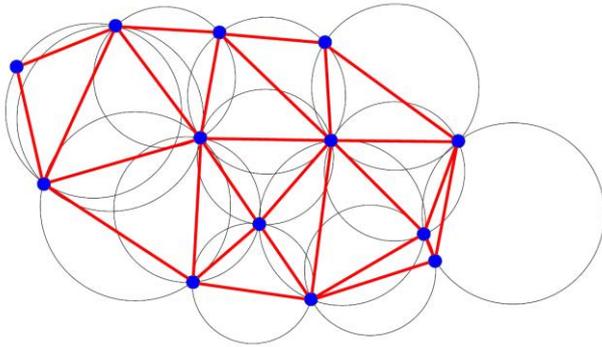


Fig. 6. A Delaunay-based triangulation with representation of empty-circle criterion.

The size and quality of triangles may still not be appropriate in the unstructured T_i formed until the current step and may cause assembling ill-conditioned discretization matrices. Therefore, in the fourth step, T_i will be refined using an optimized refinement algorithm. The mesh refinement algorithms suggest mathematical guarantees in which the quality of initial mesh structure will be improved.

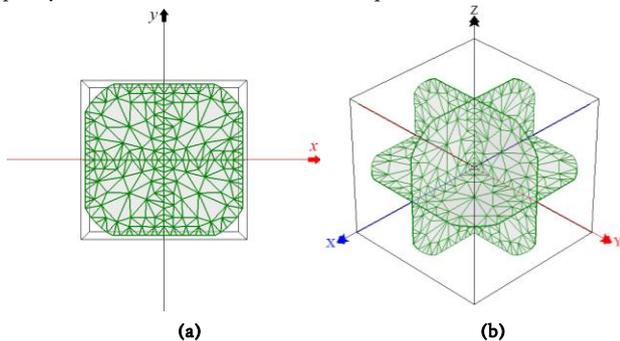


Fig. 7. Refined Delaunay triangulation for (a) a single fracture with two orthogonal intersections, and (b) three intersecting fractures, generated by developed code in the present research.

h_s is a key parameter in the present triangulation algorithm. This parameter represents the sensitivity of the algorithm to lower bounds of fractures intersections and determines the size of triangles according to the fractures length. As h_s increases, the intersections whose length is

smaller than h_s are practically reduced to a single point, but they will never be removed. Therefore, not only the connectivity pattern of fractures is maintained, but also the mesh meets the challenges discussed before. However, h_s is chosen arbitrarily by user, and is in a direct relationship with the precision of the problem. With decreasing h_s , the precision of the solution will increase but it is obvious that the computational costs will also increase due to the growth of the number of triangles. Thus, determination of the optimized h_s is a critical issue.

3. Discussion

In this section, the validation of the presented meshing algorithm and a series of sensitivity analyses on meshing parameters are discussed. At first, a scheme of numerical solution of the flow is described and then, three different examples are provided. Regular and simple geometrical structures are investigated in examples I and II to compare the outcomes of numerical flow calculations with analytical results to validate the presented algorithm. In example III, a DFN is used to conduct the sensitivity analyses.

3.1. Flow numerical solution scheme

The following assumptions are considered in this research:

- Rock matrix is impermeable.
- The flow is in a steady state.
- Two walls of each fracture are planar, smooth and parallel.
- The flow model of fluid is Newtonian.

The planar flow rate is calculated on each f_i in (x, y) with a certain aperture a_{f_i} . It is assumed that $a_{f_i} \ll L_{f_i}$, where L_{f_i} is the length of f_i . In this research, a uniform PDF has been used to determine a_{f_i} . Being dependant on the Poiseuille law, the permeability of a fracture (K_{f_i}) is obtained using Eq. 4 [50].

$$k_{f_i} = \frac{a_{f_i}^3}{12} \quad (4)$$

As given in Eq. 5, the classic equations of Darcy and conservation of mass govern the fluid flow through the fractured rock media [24].

$$\begin{cases} q_{f_i} = -\frac{1}{\mu} K_{f_i} \cdot \nabla p \\ \nabla q_{f_i} = 0 \end{cases} \quad (5)$$

$$\nabla p = \rho \cdot g \cdot \nabla h \quad (6)$$

In these equations, q_f is the average flow rate through the fracture [m^2/s], K_f is the permeability matrix of the fracture [m^3], ∇p is

pressure gradient $[Pa/m]$, ρ is the fluid density $[kg/m^3]$, g is the gravitational acceleration $[m/s^2]$ and ∇h is the hydraulic head gradient.

Any standard boundary condition can be applied to this system of equations. The boundary conditions can be either Dirichlet or Neumann. It is assumed that Γ_D and Γ_N are parts of boundaries of DFN model with Dirichlet and Neumann boundary conditions respectively; therefore, the boundary conditions are written as Eq. 7.

$$\begin{cases} h = h_D & \text{on } \Gamma_D \\ q = q_N & \text{on } \Gamma_N \end{cases} \quad (7)$$

where, h_D and q_N are boundary conditions of the hydraulic head and flow rate. For each f_i , the permeability matrix $K_{f_i} \in R^{N_{dof_i} \times N_{v_i}} \times R^{N_{dof_i} \times N_{v_i}}$ is assembled according to the number of degrees of freedom (N_{dof_i}) and vertices (N_{v_i}) of fracture. Then, the vectors $q_{f_i} \in R^{N_{dof_i} \times N_{v_i}}$ and $h_{f_i} \in R^{N_{dof_i} \times N_{v_i}}$ are considered as the vectors of the flow rate and the hydraulic head respectively. As given in Eq. 8 to 10, the total permeability matrix (K) and total vectors of flow rate (q) and hydraulic head (h) for a DFN model are derived from union of transposed local K_{f_i} , q_{f_i} and h_{f_i} respectively.

$$K = \begin{bmatrix} K_{11} & K_{12} & \dots & K_{1N} \\ K_{21} & K_{22} & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ K_{N1} & \dots & \dots & K_{NN} \end{bmatrix} \quad (8)$$

$$q = \begin{pmatrix} q_1 \\ \vdots \\ q_N \end{pmatrix} \quad (9)$$

$$h = \begin{pmatrix} h_1 \\ \vdots \\ h_N \end{pmatrix} \quad (10)$$

Where, $N = N_{v_t} \times N_{dof_t}$ and, N_{v_t} and N_{dof_t} are the total number of total vertices and degrees of freedom of the model respectively. The hydraulic head must be determined in all vertices of the model. Therefore, the total number of equations of the system is equal to N which is calculated based on the flow equilibrium conditions. To validate the presented meshing algorithm and performing the sensitivity analyses, a developed code was implemented in c# using FEM scheme with a visual three-dimensional graphical user interface to show the results. The results were calculated using the conjugate gradient (CG) method.

3.2. Example I

An example with a simple geometrical structure is provided in this section to investigate the accuracy of the linear flow calculations and to validate the meshing algorithm. The geometrical model Ω_1 includes a circular fracture with the center located at the global origin of coordinates and the radius of 5 m which is enclosed with two parallel planar boundaries by spacing of 2 m. The hydraulic head on the first boundary $H_1 = 1$ m and on the second boundary is $H_2 = 0$. The permeability of the fracture is $k = 1$ m²/s. The analytical solution for a 2×10 m rectangular slab is equal to 5 m³/s [51]. In this numerical model, the summation of the flow rate of vertices located on one of the boundaries of the model is 4.942 m³/s. The calculations demonstrate that the numerical results are in a good agreement with the analytical results. The meshing structure and the diagram of hydraulic head distribution in the direction of Y-axis is shown in Fig. 8.

3.3. Example II

According to Fig. 9, an alternative geometrical model (Ω_2) with three orthogonal planar circular fractures and three intersections in a three-dimensional space have been considered to validate the numerical solution of the fluid flow based on the presented meshing algorithm. The center of all three fractures are located at the origin of the coordinates and their radius is 71 m. Normal vectors of fractures are in the direction of X, Y and Z axes respectively. The permeability of fractures is homogeneously equal to 8.172×10^{-5} m²/s. Ω_2 is a cube with an edge length of 100 m. boundary conditions of the model have been

demonstrated in Fig. 9-a. Constant hydraulic heads of $H_1 = 1$ m and $H_2 = 0$ m are applied on the upper and lower facets of the model and lateral facets possess a constant gradient of hydraulic head. We compared the numerical solution based on the presented meshing algorithm in this example with those of Long et. al. [51]. Regarding the results, the numerical solution completely conforms to analytical results with the flow rate of 1.634×10^{-5} m³/s. The hydraulic head is constant across boundaries of the horizontal fracture and thus, no flow passes through that as the numerical solution well represents this fact. The hydraulic head in the direction of Z axis at internal vertices of fracture is equal to zero. In addition, according to the numerical solution, the resultant flow field at all whether boundary or intersection vertices of the model is equal to zero which describe the global and local conservation of mass respectively. The mesh structure and diagram of hydraulic head distribution in the Z direction is shown in Fig. 9-b and 9-c.

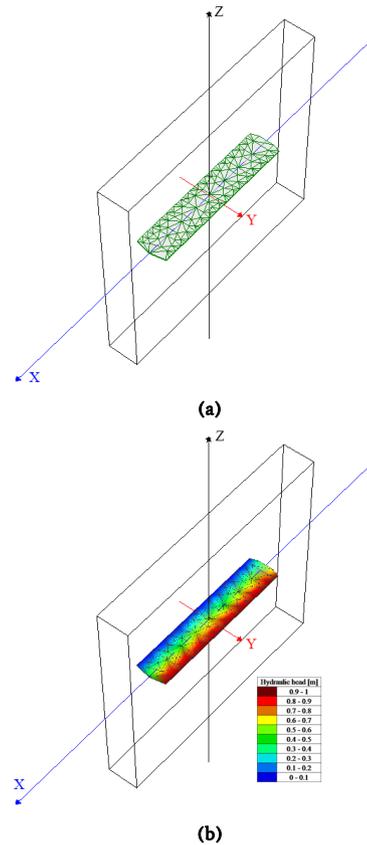


Fig. 8. (a) The meshing structure and, (b) the diagram of hydraulic head in the direction of Y axis for example I.

3.4. Example III

In this example, the numerical simulation of the fluid flow is performed for ten realizations of a DFN model ($\Omega_3^i, i = 1, \dots, 10$). In general, to decrease the effect of uncertainty in flow modeling, the numerical results for different realizations are averaged and introduced as the representative of the DFN. In this example, Ω_3^i is independently generated using the geometrical-statistical data given in Table 2, based on the technique illustrated in section 2-1. The number of fractures and intersections of Ω_3^i have been listed in Table 3. The density and viscosity of the fluid considered in this example are equal to 1000 kg/m³ and 0.001 Pa · s respectively. To figure out the effects of the meshing size (h_s) on the flow, it is changed in seven different levels. Therefore, seventy samples will be available to analyze the sensitivity of parameters. Fig. 1 shows one of the realizations for which the meshing structure and hydraulic head distribution in the Z-direction are illustrated in Fig. 10.

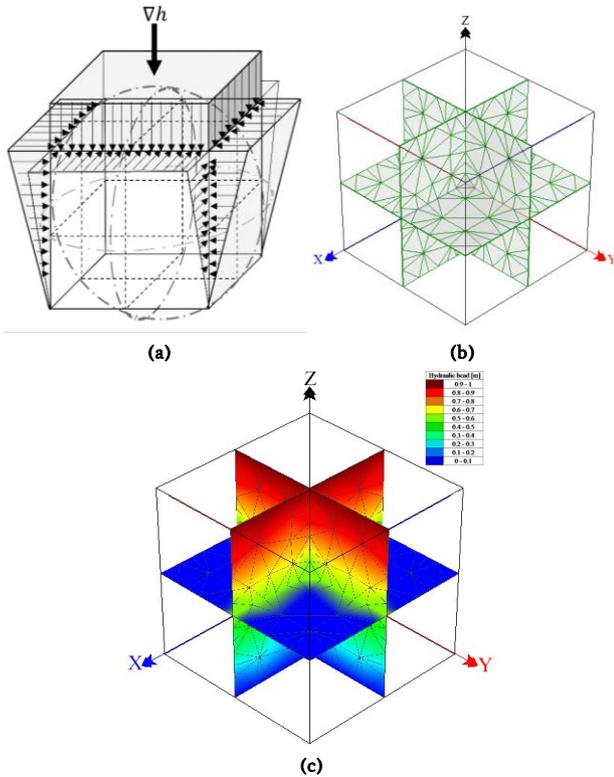


Fig. 9. (a) Boundary conditions of the hydraulic head, (b) the mesh structure and, (c) the hydraulic head distribution in the direction of Z axis for example II.

Table 2. Geometrical-statistical data of each joint set.

Parameter	Dip		Density		Length		Aperture		
	[Deg]	[Deg]	[1/m ³]	[m]	[mm]	[mm]	[mm]		
	Uniform	Fisher	Poisson	Power-law	Uniform				
	Average	k	Average	Average	α	Min	Max	Min	Max
1	70	40	45	0.2	1.78	1	10	4	12
2	30	20	135	0.12	1.78	1	10	4	12
3	80	40	135	0.1	1.78	1	10	4	12
4	45	20	315	0.15	1.78	1	10	4	12

Table 3. The number of fractures and their intersections for each realization.

Realization number	Joints number	Intersections number
1	125	313
2	115	271
3	119	331
4	130	361
5	114	324
6	117	304
7	109	269
8	122	303
9	114	270
10	111	313

In order to increase the sensitivity of the analyses, it is necessary to determine an optimized h_s . Here, the critical case is the evaluation of θ_{min} for different values of h_s . A rudimentary survey showed that if θ_{min} is unchanged, the termination of the meshing algorithm strongly affected as h_s changes. A larger θ_{min} generate a higher quality triangulation, but choosing a very large θ_{min} for a small h_s can causes instability in the meshing algorithm (lack of termination) due to forming infinite loops to refine it. Therefore, it is important to select an optimized θ_{min} for any h_s to ensure the termination of the algorithm and to befit the precision of the solution. In fact, θ_{min} is described as the value by which the termination of the algorithm is guaranteed. These results are displayed in Fig. 11, where as h_s decreases, θ_{min} decreases as

well. So, as a common result, choosing a larger h_s can generate a higher quality triangulation for Ω_3^i .

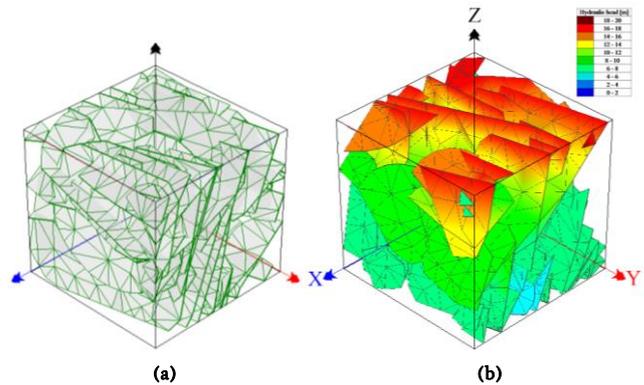


Fig. 10. (a) Mesh structure and, (b) the hydraulic head distribution in the Z-direction for DFN in example III.

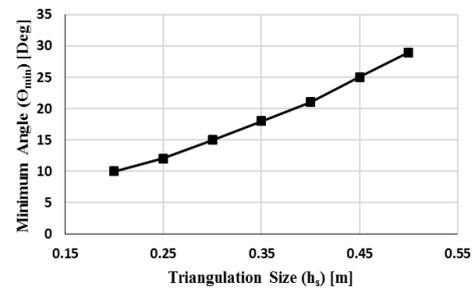


Fig. 11. Diagram of the minimum internal angle of triangles (θ_{min}) versus the meshing size (h_s).

Diagrams of the total number of vertices (N_v) and the total number of triangles (N_t) of the meshing versus h_s have been depicted in Fig. 12. However, a decreasing trend seems obvious in this figure, achieving this power-law trend is indicative of success in triangulation process of Ω_3^i . Also, with increasing h_s , N_v and N_t are led to constant numbers. Earlier is the number of vertices from union of the intersection and the boundary vertices, and later is the number of Delaunay-based triangles generated by these vertices. Reducing h_s can significantly increase the number of N_v and N_t and consequently the computational cost.

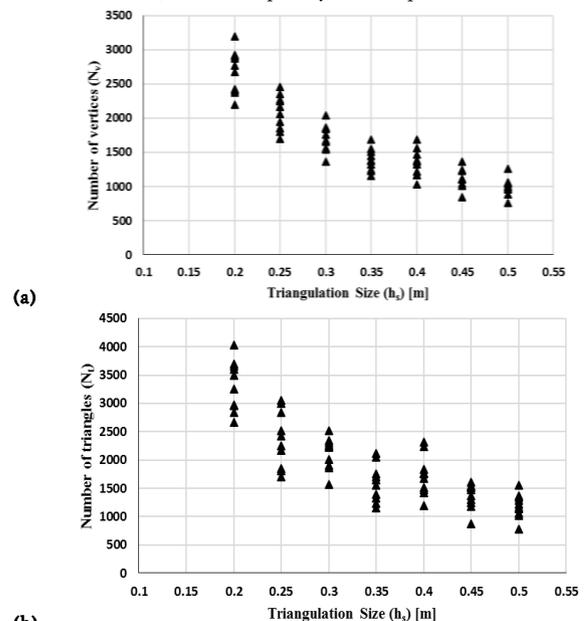


Fig. 12. (a) The total number of vertices (N_v), and (b) the total number of triangles (N_t) versus the meshing size (h_s).

Fig. 13 shows the diagram of N_v versus N_t . The relationship between

these two parameters approximately follows Eq. 11 with a correlation factor of 0.95. Because of the discrepancy in the connectivity pattern of fractures in different Ω_3^i , it is possible to generate a various number of triangles with a certain number of vertices.

$$N_v = 0.7552 N_t + 145.53 \tag{11}$$

In this research, c criterion is used to determine the precision of the solution (Eq. 12).

$$c = \frac{\|q\|_2}{N_{dof_t} \times N_{v_t}} \tag{12}$$

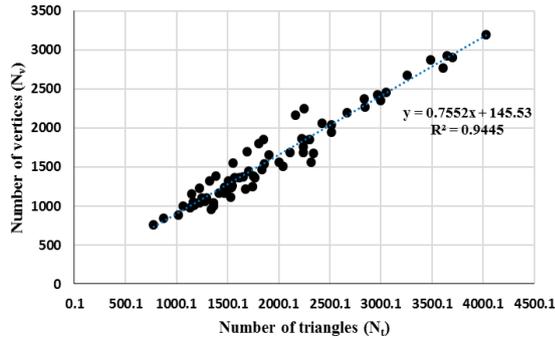
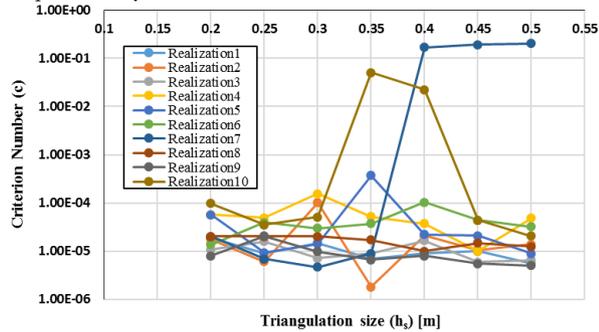
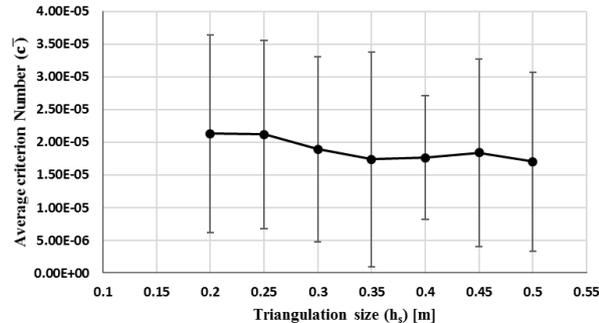


Fig. 13. Diagram of the number of vertices (N_v) versus the number of triangles (N_t) of the meshing.

Diagrams of c against h_s for all realizations and for their average are demonstrated in Fig. 14. As Fig. 14-a shows, some values of h_s cause divergence of numerical solution in some of Ω_3^i . These divergences approximately happens for $h_s > 0.3$. The diagram of Fig. 14-b has been drawn for averaged c versus h_s by ignoring the non-convergence Ω_3^i . Due to the nearly constant trend of this diagram, it seems that the convergence of the numerical solution is guaranteed for $h_s \leq 0.3$ for all the realizations. Therefore, as a secondary result, choosing a smaller h_s can computationally be more convenient.



(a)

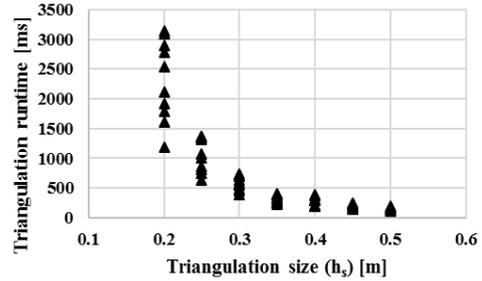


(b)

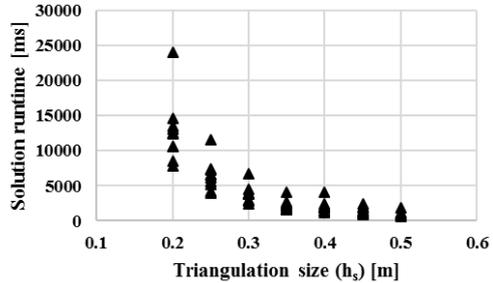
Fig. 14. Diagram of c -criterion versus the meshing size h_s for (a) all the realizations (c) and (b) average of them (c).

In Fig. 15, diagrams of runtime of the meshing algorithm and the solution scheme against h_s are depicted, respectively. In fact, N_v represents the total number of variables and the consequently the number of equations of the model for each degree of freedom. All the calculations were carried out using the same hardware system. Since, both variations of the runtime of meshing and the runtime of the solution scheme have power-law trends relative to h_s , the use of a small

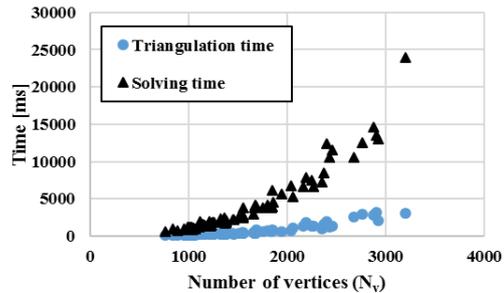
h_s can significantly increase the runtime of calculations. According to the conflicting results of Figs 11, 12, 14 and 15, it seems that $h_s = 0.3$ is an optimized meshing size for all Ω_3^i . In addition, diagram of the solution and triangulation runtimes versus N_v is shown in Fig. 15-c. According to this figure, the trend of both the runtimes follow a power-law and the triangulation runtime is dramatically less than the solution runtime for all cases. Here, the FEM solution scheme can be a standard benchmark to show how fast the present meshing algorithm is.



(a)



(b)



(c)

Fig. 15. Diagrams of the meshing size (h_s) versus (a) the meshing runtime, (b) the numerical solution runtime and, (c) diagram of both the runtimes versus the number of the vertices of the model.

4. Conclusions

In this research, a new refined Delaunay-based meshing algorithm for triangulation of geometrical structure of three-dimensional discrete fracture network was developed. This algorithm benefits from a high precision and speed, and never changes the geometrical structure and connectivity pattern of the fracture network. Also, the process of generating the discrete fractures and removing the isolated and dead-end fractures were described beside the meshing algorithm. The present algorithm was validated compared to the analytical results. In addition, a series of sensitivity analyses was conducted to determine the effect of meshing parameters on the fluid flow and to illustrate the performance of the algorithm. The provided results well represent a high quality of the meshing algorithm. It is shown that the meshing size is a key parameter in the present algorithm, therefore, a sensitivity analysis shows how to evaluate an optimized meshing size to ensure the termination of the algorithm. The authors continue to generalize the algorithm to the structures with three-dimensional (tetrahedral) meshing elements.

REFERENCES

- [1] Monteagudo J, Firoozabadi A. Control-volume method for numerical simulation of two-phase immiscible flow in two- and three-dimensional discrete-fractured media. *Water resources research*. 2004;40(7).
- [2] Ren F, Ma G, Wang Y, Li T, Zhu H. Unified pipe network method for simulation of water flow in fractured porous rock. *Journal of Hydrology*. 2017;547(80-96).
- [3] Zhu L, Liao X, Chen Z, Mu L. Development of a Coupled Discrete Dual Continuum and Discrete Fracture Model for the Simulation of Naturally Fractured Reservoirs. *SPE Reservoir Characterisation and Simulation Conference and Exhibition: Society of Petroleum Engineers*, 2017.
- [4] Klepikova MV, Borgne T, Bour O, Dreuzy JR. Inverse modeling of flow tomography experiments in fractured media. *Water Resources Research*. 2013;49(11):7255-65.
- [5] Lei Q, Latham J-P, Tsang C-F. The use of discrete fracture networks for modelling coupled geomechanical and hydrological behaviour of fractured rocks. *Computers and Geotechnics*. 2017;85(151-76).
- [6] Wang M, Kulatilake P, Panda B, Rucker M. Groundwater resources evaluation case study via discrete fracture flow modeling. *Engineering Geology*. 2001;62(4):267-91.
- [7] Xu C, Dowd P. A new computer code for discrete fracture network modelling. *Computers & Geosciences*. 2010;36(3):292-301.
- [8] Wang K, Peng X, Du Z, Haghghi M, Yu L. An Improved Grid Generation Approach for Discrete Fracture Network Modelling Using Line Fracture Concept for Two-Phase Flow Simulation. *SPE Asia Pacific Oil & Gas Conference and Exhibition: Society of Petroleum Engineers*, 2016.
- [9] Nejati M, Paluszny A, Zimmerman RW. A finite element framework for modeling internal frictional contact in three-dimensional fractured media using unstructured tetrahedral meshes. *Computer Methods in Applied Mechanics and Engineering*. 2016;306(123-50).
- [10] Parashar R, Reeves DM. On iterative techniques for computing flow in large two-dimensional discrete fracture networks. *Journal of computational and applied mathematics*. 2012;236(18):4712-24.
- [11] Karimi-Fard M, Firoozabadi A. Numerical Simulation of Water Injection in Fractured Media Using the Discrete Fractured Model and the Galerkin Method. *SPERE* 6 (2): 117–126. SPE-83633-PA. DOI: 10.2118/83633-PA, 2003.
- [12] Zhang Q-H, Yin J-M. Solution of two key issues in arbitrary three-dimensional discrete fracture network flow models. *Journal of hydrology*. 2014;514(281-96).
- [13] Gallager RG. *Discrete stochastic processes*: Springer Science & Business Media, 2012.
- [14] Wu Q, Kulatilake P, Tang H-m. Comparison of rock discontinuity mean trace length and density estimation methods using discontinuity data from an outcrop in Wenchuan area, China. *Computers and Geotechnics*. 2011;38(2):258-68.
- [15] Priest S, Hudson J. *Discontinuity spacings in rock*. *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*: Elsevier, 1976. p. 135-48.
- [16] Zhan J, Chen J, Xu P, Han X, Chen Y, Ruan Y, et al. Computational framework for obtaining volumetric fracture intensity from 3D fracture network models using Delaunay triangulations. *Computers and Geotechnics*. 2017;89(179-94).
- [17] Dershowitz W, Einstein H. Characterizing rock joint geometry with joint system models. *Rock mechanics and rock engineering*. 1988;21(1):21-51.
- [18] Han X, Chen J, Wang Q, Li Y, Zhang W, Yu T. A 3D fracture network model for the undisturbed rock mass at the Songta dam site based on small samples. *Rock Mechanics and Rock Engineering*. 2016;49(2):611.
- [19] Jin W, Gao M, Yu B, Zhang R, Xie J, Qiu Z. Elliptical fracture network modeling with validation in Datong Mine, China. *Environmental Earth Sciences*. 2015;73(11):7089.
- [20] Fardin N, Stephansson O, Jing L. Scale effect on the geometrical and mechanical properties of rock joints. *10th ISRM Congress: International Society for Rock Mechanics*, 2003.
- [21] Tsang Y. Usage of “equivalent apertures” for rock fractures as derived from hydraulic and tracer tests. *Water Resources Research*. 1992;28(5):1451-5.
- [22] Moradi M, Shamloo A, Asadbegi M, Dezfuli AD. Three dimensional pressure transient behavior study in stress sensitive reservoirs. *Journal of Petroleum Science and Engineering*. 2017;152(204-11).
- [23] Liu P, Yao J, Couples GD, Ma J, Huang Z, Sun H. Modeling and simulation of wormhole formation during acidization of fractured carbonate rocks. *Journal of Petroleum Science and Engineering*. 2017;154(284-301).
- [24] Koudina N, Garcia RG, Thovert J-F, Adler P. Permeability of three-dimensional fracture networks. *Physical Review E*. 1998;57(4):4466.
- [25] Olson JE. Joint pattern development: Effects of subcritical crack growth and mechanical crack interaction. *Journal of Geophysical Research: Solid Earth*. 1993;98(B7):12251-65.
- [26] Andersson J, Dverstorp B. Conditional simulations of fluid flow in three-dimensional networks of discrete fractures. *Water Resources Research*. 1987;23(10):1876-86.
- [27] Rutqvist J, Leung C, Hoch A, Wang Y, Wang Z. Linked multicontinuum and crack tensor approach for modeling of coupled geomechanics, fluid flow and transport in fractured rock. *Journal of Rock Mechanics and Geotechnical Engineering*. 2013;5(1):18-31.
- [28] Beyabanaki SAR, Jafari A, Biabanaki SOR, Yeung MR. A coupling model of 3-D discontinuous deformation analysis (3-D DDA) and finite element method. *AJSE*. 2009;34(2B):107-19.
- [29] Jing L, Ma Y, Fang Z. Modeling of fluid flow and solid deformation for fractured rocks with discontinuous deformation analysis (DDA) method. *International Journal of Rock Mechanics and Mining Sciences*. 2001;38(3):343-55.
- [30] Latham J-P, Xiang J, Belayneh M, Nick HM, Tsang C-F, Blunt MJ. Modeling stress-dependent permeability in fractured rock including effects of propagating and bending fractures. *International Journal of Rock Mechanics and Mining Sciences*. 2013;57(100-12).
- [31] Guo L, Latham J-P, Xiang J, Lei Q. A numerical investigation of fracture pattern and fracture aperture development in multi-layered rock using a combined finite-discrete element method. *47th US Rock Mechanics/Geomechanics Symposium: American Rock Mechanics Association*, 2013.

- [32] Elsworth D. A hybrid boundary element-finite element analysis procedure for fluid flow simulation in fractured rock masses. *International journal for numerical and analytical methods in geomechanics*. 1986;10(6):569-84.
- [33] Mustapha H, Dimitrakopoulos R, Graf T, Firoozabadi A. An efficient method for discretizing 3D fractured media for subsurface flow and transport simulations. *International Journal for Numerical Methods in Fluids*. 2011;67(5):651-70.
- [34] Hyman JD, Gable CW, Painter SL, Makedonska N. Conforming Delaunay triangulation of stochastically generated three dimensional discrete fracture networks: a feature rejection algorithm for meshing strategy. *SIAM Journal on Scientific Computing*. 2014;36(4): A1871-A94.
- [35] Berrone S, Fidelibus C, Pieraccini S, Scialò S. Simulation of the steady-state flow in discrete fracture networks with non-conforming meshes and extended finite elements. *Rock mechanics and rock engineering*. 2014;47(6):2171.
- [36] Maryška J, Severýn O, Vohralík M. Numerical simulation of fracture flow with a mixed-hybrid FEM stochastic discrete fracture network model. *Computational Geosciences*. 2005;8(3):217-34.
- [37] Mustapha H, Mustapha K. A new approach to simulating flow in discrete fracture networks with an optimized mesh. *SIAM Journal on Scientific Computing*. 2007;29(4):1439-59.
- [38] Erhel J, De Dreuzy J-R, Poirriez B. Flow simulation in three-dimensional discrete fracture networks. *SIAM Journal on Scientific Computing*. 2009;31(4):2688-705.
- [39] Mustapha H. A Gabriel-Delaunay triangulation of complex fractured media for multiphase flow simulations. *ECMOR XIII-13th European Conference on the Mathematics of Oil Recovery* 2012.
- [40] Karimi-Fard M, Durlofsky LJ, Aziz K. An efficient discrete fracture model applicable for general purpose reservoir simulators. *SPE Reservoir Simulation Symposium: Society of Petroleum Engineers*, 2003.
- [41] Li S, Xu Z, Ma G, Yang W. An adaptive mesh refinement method for a medium with discrete fracture network: The enriched Persson's method. *Finite Elements in Analysis and Design*. 2014;86(41-50).
- [42] Pichot G, Erhel J, de Dreuzy J-R. A generalized mixed hybrid mortar method for solving flow in stochastic discrete fracture networks. *SIAM Journal on scientific computing*. 2012;34(1): B86-B105.
- [43] Hu M, Rutqvist J, Wang Y. A practical model for fluid flow in discrete-fracture porous media by using the numerical manifold method. *Advances in Water Resources*. 2016;97(38-51).
- [44] Benedetto MF, Berrone S, Scialò S. A globally conforming method for solving flow in discrete fracture networks using the virtual element method. *Finite Elements in Analysis and Design*. 2016;109(23-36).
- [45] Meyers ATG, Priest SD. Generating discontinuity orientation data for use in probabilistic models for modelling excavations in rock. *ISRM International Symposium: International Society for Rock Mechanics*, 2000.
- [46] Vermilye JM, Scholz CH. Relation between vein length and aperture. *Journal of structural geology*. 1995;17(3):423-34.
- [47] Jing L. A review of techniques, advances and outstanding issues in numerical modelling for rock mechanics and rock engineering. *International Journal of Rock Mechanics and Mining Sciences*. 2003;40(3):283-353.
- [48] Cheng S-W, Dey TK, Shewchuk J. *Delaunay mesh generation*: CRC Press, 2012.
- [49] Shewchuk JR. Delaunay refinement algorithms for triangular mesh generation. *Computational geometry*. 2002;22(1-3):21-74.
- [50] Baca R, Arnett R, Langford D. Modelling fluid flow in fractured-porous rock masses by finite-element techniques. *International Journal for Numerical Methods in Fluids*. 1984;4(4):337-48.
- [51] Long J, Gilmour P, Witherspoon PA. A model for steady fluid flow in random three-dimensional networks of disc-shaped fractures. *Water Resources Research*. 1985;21(8):1105-15.