

Advances in Geocomputing

Bearbeitet von
Huilin Xing

1. Auflage 2009. Buch. xviii, 325 S.

ISBN 978 3 540 85877 5

Format (B x L): 15,5 x 23,5 cm

Weitere Fachgebiete > Geologie, Geographie, Klima, Umwelt > Geologie und
Nachbarwissenschaften > Geoinformatik

Zu Inhaltsverzeichnis

schnell und portofrei erhältlich bei


DIE FACHBUCHHANDLUNG

Die Online-Fachbuchhandlung beek-shop.de ist spezialisiert auf Fachbücher, insbesondere Recht, Steuern und Wirtschaft. Im Sortiment finden Sie alle Medien (Bücher, Zeitschriften, CDs, eBooks, etc.) aller Verlage. Ergänzt wird das Programm durch Services wie Neuerscheinungsdienst oder Zusammenstellungen von Büchern zu Sonderpreisen. Der Shop führt mehr als 8 Millionen Produkte.

II. 3D Mesh Generation in Geocomputing

Huilin Xing,¹ Wenhui Yu^{1,2} and Ji Zhang¹

¹The University of Queensland, Earth Systems Science Computational Centre, St. Lucia, Brisbane, QLD 4072, Australia

²Department of Engineering Mechanics, Dalian University of Technology, Dalian, China

Abstract Mesh generation has been widely used in engineering computing, but seems to be relatively “new” for geoscience community. This paper firstly lists such relevant progresses of mesh generation for the engineering computing and then discusses the possibility for applying/extending them to geoscience computing (i.e. geocomputing). For geoscience, the available input data are normally a large quantity of point data in the 3D space rather than the defined shapes and dimensions with reasonable tolerances provided by the industrial designers, thus quite different from the engineering cases. To deal with such geoscience data, this paper briefly introduces the relevant progresses on geometrical modeling, hexahedral and tetrahedral shaped mesh generation, and then focuses on the applying and/or extending the related methods to generate all hexahedral/tetrahedral shaped meshes in 3D for geoscience purposes, which is described through the different practical application examples, such as the all-hexahedral shaped mesh generation for a fracture dominated reservoir system, the South Australia interacting fault system and the entire earth model without or with the simplified/practical plate boundaries; and all-tetrahedral shaped mesh generation for a multi-layer underground geological model, and visualizing and meshing with the microseismicity data recorded during a hydraulic stimulation process in a geothermal reservoir.

1 Introduction

With the rapid development of supercomputers, high-performance computing based simulation is internationally recognized as paradigm shift that offers an outstanding opportunity for advancement to better understand and quantify the earth systems science, materials science and engineering etc. Especially for the geoscience, short-term idealized experiments and field site observations have helped people to understand the relevant phenomena, while the prediction and risk assessment of complex earth systems may be better to be achieved numerically, that includes the finite element method (FEM), the finite difference method (FDM). FDM normally requires the simplified regular square/cubic mesh (even with overlap), which is much easier to deal with. Therefore, the mesh generation for FEM analysis is focused here. Several commercial FEM software packages are available and widely used in the practical for industrial engineering design and analysis, such as ABAQUS, ANSYS, ADINA, MSC Software and LS-Dyna3D. Following the above successful stories in the mechanical and civil engineering computing, the finite element based numerical modeling of the geoscience offers an outstanding opportunity to gain an understanding of those dynamics and complex system behaviour, and to develop the scientific underpinning for geoscience, such as ground motion, geodynamics, interacting fault system, and earthquake and tsunami forecasting.

Mesh generation is a critical step before finite element analysis could be carried out, which is defined as a process of dividing a continuous physical domain into a grids (elements) for the further numerical solution. Mesh generation and optimization process may be achieved by numerous in-house and/or commercial software programs, and many researchers are still working on it. The Sandia National Laboratories' 16th International Meshing Roundtable (<http://www.imr.sandia.gov/16imr/main.html>) and the 6th Symposium on Trends in Unstructured Mesh Generation (<http://www.andrew.cmu.edu/user/sowen/meshtrends6/index.html>) was just held in 2007, which reflects the related research history, current outcomes and problems. Robert Schneiders maintains a website to provide information on mesh and grid generation: people working in the field, research groups, books and conferences (<http://www-users.informatik.rwth-aachen.de/~roberts/meshgeneration.html>); Owen also maintains a meshing research corner (see <http://www.andrew.cmu.edu/user/sowen/mesh.html>) and did a survey of unstructured mesh generation technology used in both in-house and commercial software (see <http://www.andrew.cmu.edu/user/sowen/survey/index.html>); Xing and Mora (2003) did a similar survey but focused on the technologies mostly relevant to mesh generation of crustal fault systems. Based on these surveys, surface domains may be subdivided

into triangle or quadrilateral shaped mesh and volumes may be subdivided primarily into tetrahedral or hexahedral shaped mesh by using various of software, where the following algorithms are mostly used: Octree (e.g. Frey et al. 1994; Schneiders and Bunten 1995, 1996; Schneiders 1996, 1997; Shephard and Marcel 1991, 1992; Tu and O'Hallaron 2004; Yerry and Shephard 1984), Delaunay (e.g. Baker 1989; Borouchaki and Lo 1995; Borouchaki et al. 1996; Borouchaki and George 1997; Borouchaki and Frey 1998; Borouchaki et al. 2000; Du and Wang 2003; George et al. 1991; George and Seveno 1994; George 1997; Joe 1991a, b, c, 1995; Lee and Schacter 1980; Shewchuk 2002; Weatherill and Hassan 1994; Wright and Alan 1994), advanced front algorithm (e.g. Lau and Lo, 1996; Lee and Lo 1994; Lee and Hobbs 1999; Lee 2003; Lee and Lee 2002, 2003; Lo 1991a, b, 1992; Lohner 1996a, b, c; Lohner and Cebal 2000; Lohner and Eugenio 1998; Owen et al. 1999; Owen and Saigal 2000; Yamakawa and Shimada 2003), and sweeping/mapping method (e.g. Cheng and Li 1996; Knupp 1998, 1999; Lai et al. 2000; Scott et al. 2005; Staten et al. 1998, 2005). A brief summary is listed in Table 1. Automatic generation of triangle and quadrilateral shaped mesh in 2D and tetrahedral shaped mesh in 3D for a normal continuous domain is already quite mature, but for a high quality hexahedral shaped mesh, the automatic mesh generation of a general 3D physical domain is still not available, which is mainly achieved through the case-by-case trial-and-error techniques. Moreover, for a discontinuous complex domain, further research for automatic mesh generation of tetrahedron/hexahedron shaped meshes is still required.

Table 1 Various algorithms used in mesh generation and their advantages/disadvantages

	Delaunay	AFT	Quadtree/ Octree	Indirect/ Sweeping/ Mapped Meshing
Supported Element Type	Triangle/ Tetrahedron	Triangle/ Tetrahedron/ Quadrangle/ Hexahedron	Triangle/ Tetrahedron/ Quadrangle/ Hexahedron	Hexahedron
Algorithm Efficiency	$O(N \log(N))$	$O(N \log(N))$	$O(N \log(N))$	
Element Quality	Normally good	Normally good	Good except boundary	Normally good, but not always achievable

(Continued)

Table 1 (Continued)

	Delaunay	AFT	Quadtree/ Octree	Indirect/ Sweeping/ Mapped Meshing
Adaptivity	Yes	Yes	Yes	Case by case
Automatic generation	Yes	Yes	Yes	Case by case
Known problems	Boundary recovery/ Sliver De-composition	Convergence problem	Boundary fitting	Sometimes not achivable
Commecial Software/ Public Code	ANSYS ^a / Qhull ^b / Triangle ^c	ANSYS/GID ^d / Hypermesh ^e	ICEM CFD ^f / MEGA ^g / MESH ^h /QMG ⁱ	ANSYS/ GID/ Hypermesh/ CUBIT ^j

^asee <http://www.ansys.com> for more information.

^bsee <http://www.qhull.org> for more information.

^csee <http://www.cs.cmu.edu/~quake/triangle.html> for more information.

^dsee <http://gid.cimne.upc.es> for more information.

^esee http://www.altair.com/software/hw_hm.htm for more information.

^fsee <http://www.icemcfd.com/> for more information.

^gsee <http://www.scorec.rpi.edu/> for more information.

^hsee http://www.synopsys.com/products/tcad/mesh_ds.html for more information.

ⁱsee <http://simon.cs.cornell.edu/Info/People/vavasis/qmg-home.html> for more information.

^jsee <http://cubit.sandia.gov> for more information.

The existing commercial and industrial strength in-house graphics software as above are, on the whole, designed for the mechanical and civil engineering industry (such as the automobile industry). In these industries, the domain normally has defined shapes and dimensions with reasonable tolerances provided by the designers. Many applications use a “bottom-up” approach to mesh generation. Vertices are firstly meshed, followed by curves, then surfaces and finally solids. The input for the subsequent meshing operation is the result of the previous lower dimension meshing operation. For an example, nodes are firstly placed at all vertices of the geometry. Nodes are then distributed along the geometric curves. The result of the curve meshing process provides input of a surface meshing algorithm, where a set of curves define a closed set of surface boundaries. Decomposing the surface into well-shaped triangles or quadrilaterals is the next stage of the meshing process. Finally, if a solid model is provided as the geometric domain, a set of meshed surfaces defining a closed volume is provided

as input of a volume mesher for automatic generation of tetrahedra, hexahedra or mixed element types (see Owen's survey as indicated above).

However, difficulties have been encountered by using the existing commercial graphics packages (including the industrial strength in-house software) to construct a computational model of a certain geoscience case, such as an interacting fault system. Specifically, it is very difficult and time-consuming to use the existing engineering-focused software to construct practical 3D models with complex fault geometries, which are given by the converted fault data from digital images or a serial of point sets. This is mainly because the existing commercial graphics software are, on the whole, designed for the mechanical and civil engineering industry (such as the automobile industry) as described above. In these industries, the domain normally has defined shapes and dimensions with reasonable tolerances provided by the designers, whereas, in the cases of geoscience, these are usually specified by a large quantity of point data, such as the fault data. Therefore, special techniques are required for treatment of the different cases from the geoscience community. In addition, mesh generation seems to be quite "new" for the geoscience community despite it is widely used in the engineering computing. To follow the successful stories in the mechanical/civil engineering computing etc., a preliminary introduction/training is necessary to ensure such successful engineering-focused mesh generation procedures and relevant software be helpful and applicable to the geoscience field.

Automatic generation of triangle and quadrilateral shaped mesh in 2D is already quite mature, thus the 3D cases are focused here. To generate meshes containing faults (discontinuity) described by a large amount of point data set (including digital images) in 3D space, tetrahedral shaped elements seem to be more easily achieved automatically by using the Delaunay algorithm (the mostly common one) and the advancing front technique (AFT). While the conventional Delaunay algorithm is not ideal since it is difficult to guarantee that nodal points are exactly located at the fault/boundary interfaces specified by the input point data. The AFT may generate a mesh with nodes located exactly at fault interfaces, although very few codes are available with this algorithm for automatic mesh generation due to a few of its limitations. Hexahedral shaped elements provide an alternative to tetrahedral shaped meshes. So far, there is no software tool available to automatically generate a high quality hexahedral mesh for a general 3D physical domain including the fault system, despite a lot of outcomes have been achieved, such as CUBIT (<http://sass1693.sandia.gov/cubit/>).

In summary, mesh generation is well developed and widely used in the engineering field as above, but it seems to be quite "new" for the geoscientists, thus the successful experiences and outcomes in the other field would

be quite helpful and useful for the geoscience community. Here, we focus on applying and/or extending such relevant successful experiences and methods to generate all hexahedral/tetrahedral shaped mesh in 3D for the geoscience purposes through several different practical application examples.

2 Geometrical Modeling

With the rapid development of advanced digital image technology and its application in the earth sciences, more and more image information of the Earth is available and may be used to build computational models and even to validate numerical results. However, the available input data for geoscience are quite different from the engineering case as described above; it is a large quantity of point data (including digital images) rather than defined shapes and dimensions with reasonable tolerances provided by the industrial designers. Thus there are a number of challenges that must be overcome, such as how to obtain and use geological point data set to construct the computational model and in particular generate usable meshes (elements) for the further finite element analysis. Such a pre-processing normally includes geometrical modeling and mesh generation. Here geometrical modeling aims to construct and manage the related geological data, and it can also be applied to reconstruct the tectonic evolution of the whole Earth, or a specified region such as the western Mediterranean since the Oligocene (e.g. Rosenbaum et al. 2002). The task of geometrical modeling here concentrates on the geological model construction, which aims at editing and managing the tectonic data, and constructing 2D or 3D geological surface/solid model involving faults and/or plate boundaries. A specially-purposed tectonic CAD/Database module was developed within the CHIKAKU system (Kanai et al. 2000; Xing et al. 2001), which aims to manage the data and construct a computational model for an interacting fault system. It is composed of two subsystems: tectonic database and CAD. The main purpose is to construct 2D and/or 3D solid models including faults and plate boundaries and to output the solid model of the specified region directly for the CHIKAKU Mesh (Xing et al. 2001, 2007a) and/or in the standard IGES data format for easily interfacing with other software packages (such as I-DEAS and Patran) for further mesh generation. The developed geometrical modeling module includes the following three aspects: (1) Data input: The data may include (a) the underground structural data of stratum boundary point, stratum borderline, stratum and plate/fault, which could be available through the natural and/or man-made earthquake data inversion, drilling well and advanced digital image etc.; (b) observational data: hypocentral distribution and distribution

of the distortion and (c) reference data for coastlines, the administration field, rivers, the Earth's surface and the ocean floor. (2) Data management and editing: conversion of digital images and other data to the necessary formats for constructing the fault and plate models, visualization and editing of the above data; (3) Geometrical modeling: construction and editing of the lines and curves defining faults and plate boundaries, construction of the parametric surfaces using the above curves, construction and editing of the solid models using the generated surfaces, editing and output of a solid model of a specified region. The above function may be also partly or totally achieved by other in-house or commercial software such as those specified in the above surveys with the similar procedure. Upon completion of the geometrical modeling, the mesh generation and optimization are carried out together with the specification of loading, boundary conditions and material properties and so on.

3 Hexahedral Mesh Generation

3.1 Introduction

There is the often-held position that quadrilateral and hexahedral shaped elements have superior performance over tetrahedral shaped elements when comparing an equivalent number of degrees of freedom, and also more suitable for nonlinear finite element analysis in some cases. For finite element analysis of incompressible or nearly incompressible nonlinear behavior, such as the large deformation of the Earth in 3D, it seems to be necessary to use a hexahedral mesh rather than a tetrahedral mesh to obtain sufficiently accurate results. The hexahedral shaped mesh generation is crucial for the finite element community, but the automatic generation of such a high quality all-hexahedral mesh for a general three-dimension domain is still not available, especially for meshing a large scale complex geometry containing faults (discontinuities).

Due to difficulties in automatic generation of such a high quality all-hexahedral mesh, several methods on special classes of geometry have been proposed and become the easier ways to go, such as (1) the Octree method (e.g. Schneiders et al. 1996; Loic 2001), which generates small cubes inside the geometric model and generates a mesh by mapping the surfaces of the boundary cubes onto the surfaces of the geometric model; Normally, it is difficult to fit the boundary well, thus its application is quite limited; (2) the mapped method, which firstly decomposes the whole geometry to one or several meshable blocks before meshing (Shin and Sakurai 1996; Taghavi 2000; Calvo and Idelsohm 2000); In addition, a shape recognition and boundary fitting based method is also proposed (Takahashi

and Shimizu 1991; Chiba et al. 1998), which employs a unique shape-recognition technique to change a geometric model into an approximate one consisting of straight lines. Boundary fitting maps small cubes that are generated by dividing the approximate model, onto the geometric and generate hexahedral meshes. However, this sometimes can not be always successfully applied in case of some complicated models, such as (a) If the geometric model contains surfaces which has three or fewer edges, a recognition model that is topologically equal to the geometric model cannot be generated; and (b) If the assigned edge directions are not correct, a recognition model cannot be generated even if the geometric model is topologically correct. This method has then been improved by automating the model-editing task using feature line extraction (Hariya et al. 2006), but it still just works case-by-case; (3) sweeping method (e.g. Scott et al. 2005). To ensure that the geometry be meshed with all-hexahedral finite elements, sweeping requires that the geometry should be 2.5D or decomposable into 2.5D sub-geometries. It includes the following two ways: One-to-One sweeping and Many-to-One or Many-to-Many Sweeping. As for One-to-One Sweeping (Scott et al. 2005), sweeping of “One-to-One” geometry begins by identifying “source”, “target”, and connected “guiding” curves/surfaces. The source surface is then usually meshed with quadrilaterals using an unstructured scheme such as paving (Blacker and Stephenson 1991). Each guiding curves/surfaces must also be meshed with a mapped or sub-mapped mesh. The surface mesh on the source is then swept or extruded one layer at a time along the mapped mesh on the guiding curves/surfaces toward the target mesh. This type of sweep is termed as “One-to-One” because of the One-to-One correspondence between the source and target surface. Due to the One-to-One sweeping’s strict requirements, few geometry satisfy the topological constraints required to generate a swept mesh. The Many-to-One or Many-to-Many Sweeping methods have been proposed to decompose more complex geometry into 2.5D sweep “blocks” or “barrels” which then be sweepable (e.g. Lai et al. 2000; Shepherd et al. 2000; Knupp 1998; White et al. 2004). Sweepable geometries or geometries that may be decomposed into sweepable parts can be detected automatically with a fair amount of success (White et al. 2000); (4) converting from the tetrahedral shaped mesh. Normally, the tetrahedron mesh is much easier to be generated and can be converted to the hexahedron mesh, but such a kind of hexahedron mesh is normally in poor shape quality and with dramatically increased node and element numbers, thus not always acceptable for further finite element analysis. Related efforts have been attempted to re-generate the domain occupied by tetrahedral shaped mesh to the hexahedron mesh with high quality, but not always achievable (e.g. Owen et al. 1997, 2000).

The above methods are applied here to mesh the following practical geoscience problems. It is important to note that geometry decomposition is widely used as above for the hexahedral shaped mesh generation. For a continuous geometry, the actual decomposition of the geometry does not occur, only an internal characterization of sweep/mapped blocks; but for a discontinuous geometry containing faults, the discontinuous boundaries (faults) will be used as a constraint for the geometry decomposition for further mesh generation.

3.2 Fracture Dominated Reservoir System

Figure 1 shows a fault system in a certain fracture dominated gas reservoir. The faults at the upper surface are depicted as 19 curves, and the fault surfaces are straight along the vertical direction. Therefore, the sweeping method can be applied to generate the hexahedral shaped mesh for such a 2.5 dimension case. Normally, one may use all the faults at the upper surface as the constrained curves to generate the quadrilateral shaped mesh without additional geometrical operation, and then generate the hexahedral shaped mesh using the sweeping method, i.e. take the quadrilateral shaped surface mesh as front and then advance inward. It may work well for a normal continuous domain. However, for such a complicated fracture dominated reservoir system, this will make all the meshes continuous without taking all the faults as the real discontinuous boundaries (curves/surfaces) for the quadrilateral/hexahedral shaped mesh in 2D/3D, and the meshing result even fails when the constrained curves are complicated and interacted with each other, as shown in Fig. 2 with the current fault system. Therefore, the whole geometry is firstly decomposed into a number of simple meshable shaped geometries (i.e. triangular or/and quadrilateral shape) with the constraint of the fault curves as shown in Fig. 3 and put into different groups; secondly, the quadrilateral shaped mesh is generated as above but sharing the same mesh seed at the common edges (curves) between the neighbour groups; thirdly, the hexahedral shaped mesh is generated using the sweeping method for the different groups; finally, the above hexahedral meshes are assembled together with “welding” operation (i.e. node equivalent), while keeping the meshes along the faults being discontinuous, which are taken as the frictional contact interfaces in the further finite element analysis. More regular and fine meshes are used around the faults, while coarse meshes are used in the other regions. The current model consists of about 46,000 hexahedron elements with 65,000 nodes and all the 19 faults (see Fig. 4).

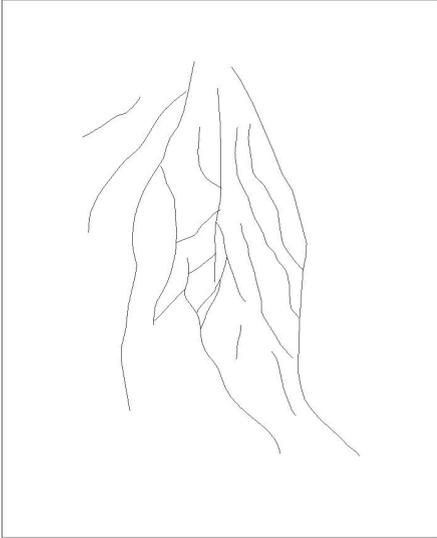


Fig. 1 The geometry and distribution of 19 faults in a certain fracture dominated gas reservoir

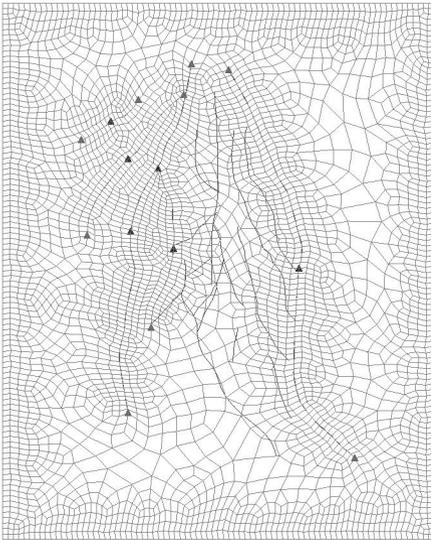


Fig. 2 A failed example of the quadrilateral shaped mesh generation on the *top* surface using all the 19 complicated faults as the *constrained curves*, because only the faults marking with *black triangular* are really involved for such an automatic operation

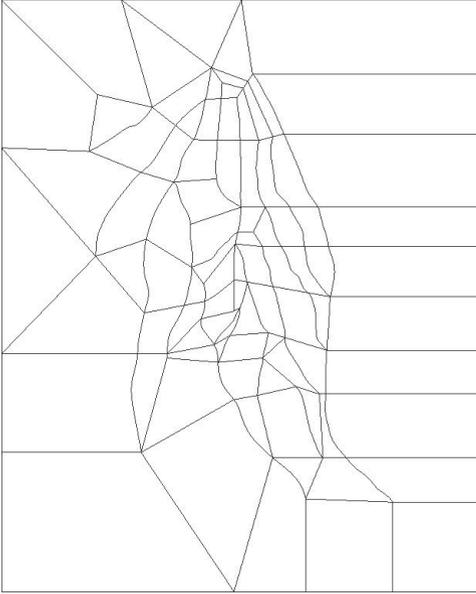
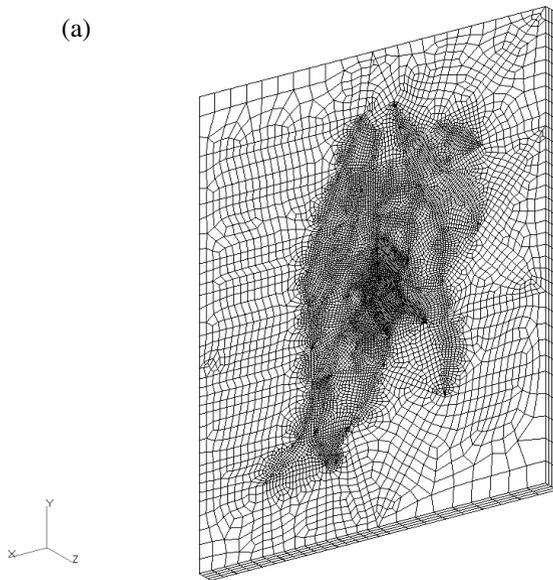


Fig. 3 The *top* surface of the above fault system is decomposed into the 2D meshable geometries for the quadrilateral shaped mesh generation with the constraint of all the 19 *fault curves*



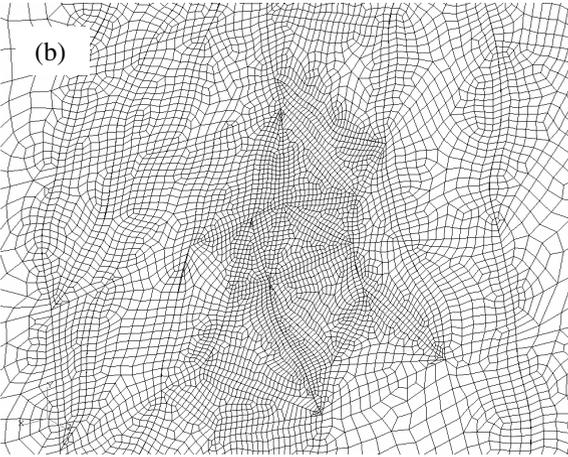


Fig. 4 The hexahedral shaped mesh generated using the sweeping method **(a)** the whole mesh and **(b)** magnification of the central part of **(a)**

The sweeping method used here is limited that the number, size, and orientation of the quadrilateral faces on opposing fronts direction should match, thus it is rarely able to resolve the unmeshed voids and real general 3D problems. Once the opposing fronts collide, the algorithm frequently has deficiencies. Many creative attempts have been made to resolve this unmeshed void left behind by plastering. For examples, since arbitrary 3D voids can be robustly filled with tetrahedral meshes, the idea of plastering in a few layers, followed by tetrahedron-meshing the remaining void was attempted (Dewhurst et al. 1995; Ray et al. 1998); Transitions between the tetrahedral and the hexahedral meshes were done with Pyramids (Owen et al. 1997) and multi-point constraints; The Geode-Template (Leland et al. 1998) provided a method of generating an all-hexahedral mesh by refining both the tetrahedral and the hexahedral meshes. However, this requires an additional refinement of the entire mesh, which resulting in node/element numbers much larger than required. In addition, the Geode-Template was unable to provide reasonable mesh quality (Staten et al. 2005).

3.3 Meshing Interacting Fault System of South Australia with Mapped Block Method

The South Australian interacting fault system was chosen as a representative intraplate fault system. This choice was based on South Australia being reasonably representative of an active fault system in Australia and the availability of sufficient data and geological expertise for this region to enable a mesh be constructed. With detailed fault data based on advanced digital images (e.g. Fig. 5a) and geological knowledge of the region, provided by Professor Mike Sandiford of Melbourne University, a few of faults along the vertical direction are not straight along the vertical direction, it is a real 3D case, thus the mapped block method rather than the sweeping method is applied here. A 3D fault geometry model within a block with dimensions of about $530 \times 350 \times 60 \text{ km}^3$ (Fig. 5b) was firstly constructed. This involved editing and smoothing the related curves/surfaces defining the faults. In order to easily generate the hexahedral mesh and specify the conditions necessary for the finite element simulation (i.e. boundary conditions and information about faults), the entire geometric model of faults was firstly divided into several different geometrical components/blocks representing components of the solid model (e.g. Fig. 6a, b, c, d, e, and f). All of them are meshable using the mapped block meshing method, and a few of them can also be meshed by sweeping method (such as that in Fig. 6c and 7c); and these blocks were then used to generate finite element meshes (e.g. see Fig. 7a, b, c, d, e and f). Finally, the hexahedral meshes generated for the different components were assembled together with “welding” operation (i.e. node equivalent), or our stick contact algorithm after meshing (Xing et al. 2007a) (see Fig. 8). As shown in Fig. 8, more regular and fine meshes are used around the faults, while coarse meshes are used in the other regions. This approach enables more accurate computational results be obtained with reasonable finite computational resources. The discretised model after optimization currently includes 504,471 nodes and 464,620 8-node hexahedron elements together with 9 contact interfaces (i.e. faults). It was further analyzed by using our finite element code (Xing and Mora 2006).

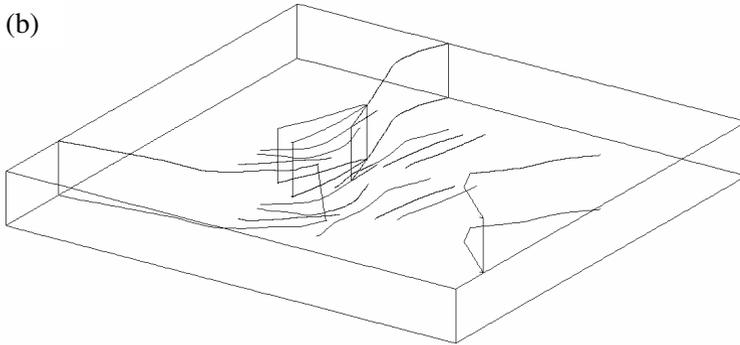
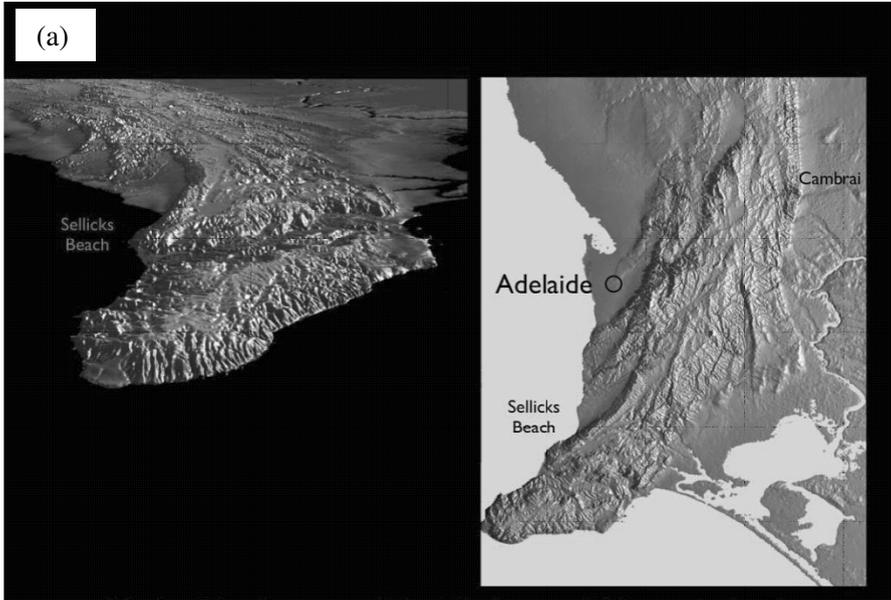


Fig. 5 The South Australian interacting fault system **(a)** Image of the region of South Australia being considered; **(b)** 3-D fault model to be analyzed

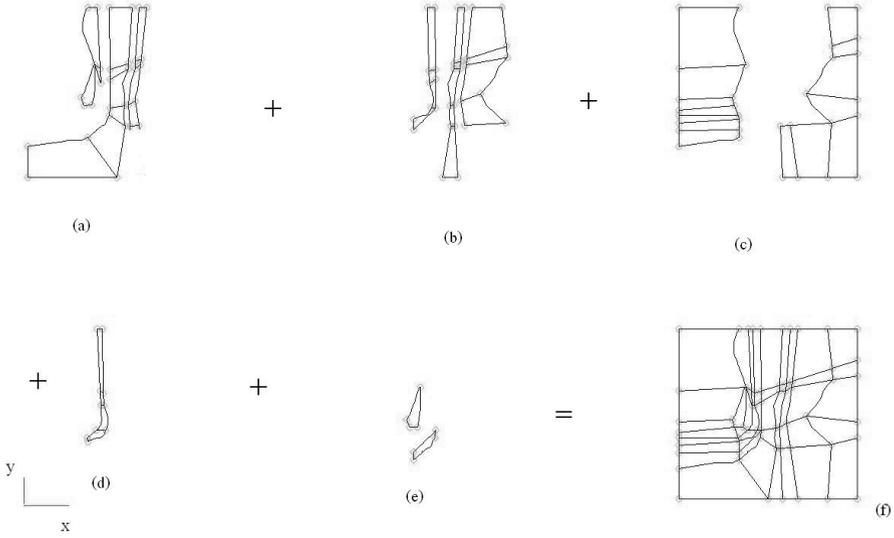
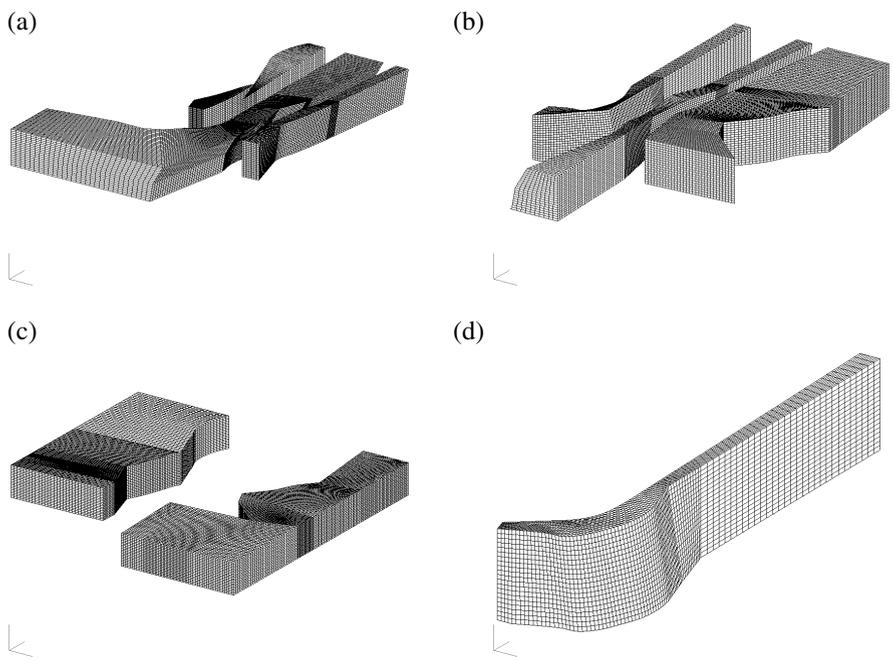


Fig. 6 The entire geometric model of SA fault system is decomposed into several different geometrical components/blocks and shown in the XY plane



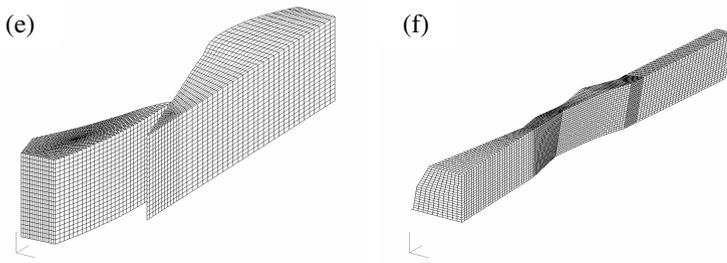


Fig. 7 (a, b, c and d) The hexahedral shaped mesh generated respectively from the components shown in Fig. 6a, b, c and d; (e) The *left* component of that in (a); (f) The *middle* component of that in (b)

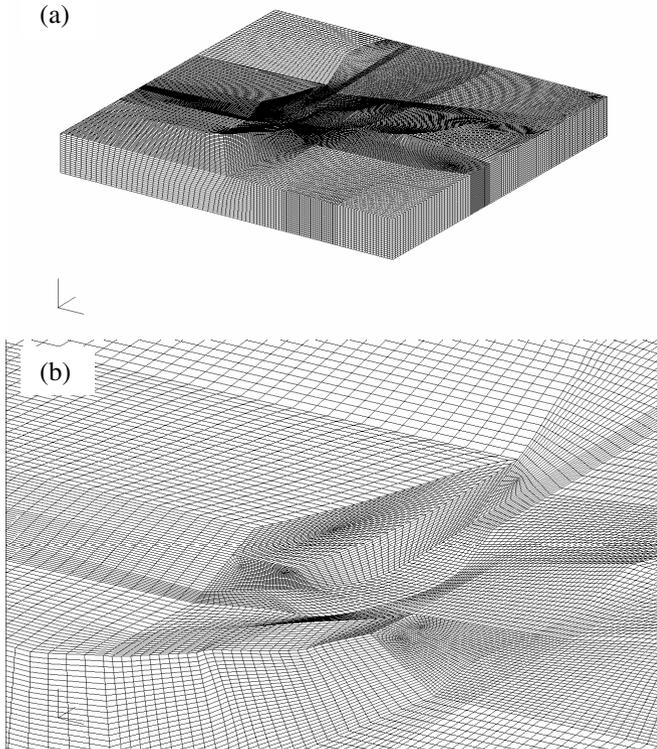


Fig. 8 (a) The total mesh generated after assembling all the components together; (b) magnification of the *central part* of the *upper surface* of (a)

3.4 All Hexahedron Mesh Generation for a Whole-Earth Model

Mesh generation for a certain region in the reservoir and the fault system scale is described above, but sometimes the whole-earth model is highly required, such as for the tide deformation/stress analysis, deformable whole-earth rotation, the whole-earth system analysis and the interaction of a planetary system with the earth. Geophysical earth-models have advanced from the simplest Guttenburg-Bullen Model to the more complex Preliminary Reference Earth Model (PREM) (Dziewonski and Anderson 1981) and current 3D models including plate/fault boundaries. This provides more opportunities for the further analysis with a more accurate earth model, while it is a great challenge for the mesh generation before the finite element analysis could be carried out. Yin-Yang grid (Kagayama et al. 2004) used in the mantle convection simulation of the whole-earth, which is easily generated, but the meshes with Yin-Yang method are partly overlapped and not suitable for the finite element modeling. Our efforts on all-hexahedral shaped mesh generation of the whole-earth of model without/with considering the real plate boundaries are introduced here.

3.4.1 The PREM whole-Earth model

The PREM model (Dziewonski and Anderson 1981) is a widely applied whole-earth model. It is composed of the inner core, the outer core, the mantle, the transition zone and the crust. For simplicity, we simplify it to a four-layered geophysical earth-model: the inner core, the outer core, the mantle and the crust (including the transition zone), as shown in Fig. 9. For convenience, the mapped block technique is applied here to decompose the whole spherical earth to different meshable blocks (groups). Each layer of the above whole-Earth model consists of 6 groups, thus 24 groups are generated in the whole-Earth model (Fig. 9), the model construction and mesh generation are thus much more simplified. The whole-Earth is discretised into 44,602 nodes and 43,008 hexahedron elements for the continuous case. For simplicity, both the same and the different material parameters can be assigned in each layer (but varying amongst the different layers) for the further finite element analysis.

For a certain case for simply analyzing the fault effects, such as the tidal deformation of the entire earth with a discontinuous outer layer (Xing et al. 2007b), a simplified fault (i.e. a discontinuous seismogenic interface) is enough, which can be assumed to exist in the outer layer between the groups g_1 and g_{1_3} as shown in Fig. 9. But the practical plate boundaries with the detailed geometry and fault properties (i.e. frictional contact

parameters along the plate boundaries) are necessary for such as earthquake dynamics analysis.

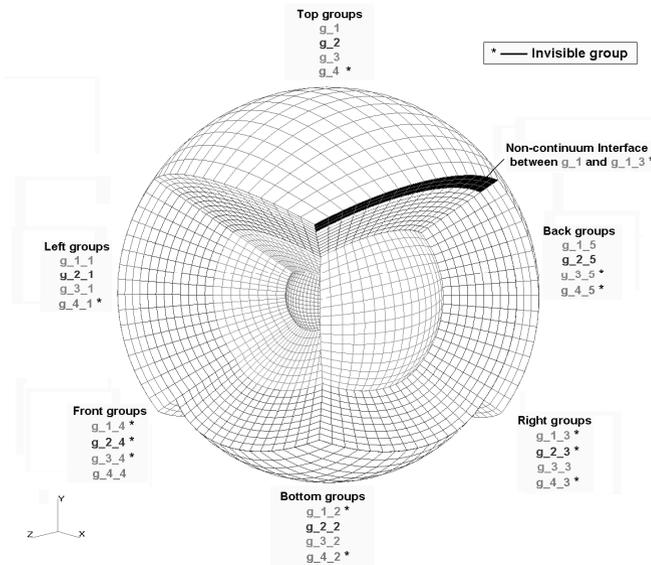


Fig. 9 The entire geophysical Earth-model to be analysed. It is composed of four layers (from the *inside* to the *outside*): the *inner core*, the *outer core*, the *mantle* and *outer layer* (crust), and each layer consists of 6 groups

3.4.2 The Whole-Earth Crust with Plate Boundaries

As described above, various earth models are available now. Here, the plate boundary data from the Plates Project at Texas University is applied to construct the whole-earth discontinuous crustal model (Fig. 10. See the detailed data at <http://www.ig.utexas.edu/research/projects/plates/index.htm>). It seems to be impossible to mesh the crustal layer with such complicated plate boundary geometries by directly using the above meshing methods.

For simplification, we firstly construct and edit the fault data in the longitude-latitude plane coordinate system (i.e. with the map Mercator projection), and create the curves with the input point data of the plate boundaries, then divide it into three following domains respectively: Domain A, ranging for the longitude $[-180^\circ, 180^\circ]$ and the latitude $[-90^\circ, -85^\circ]$; Domain B: longitude $[-180^\circ, 180^\circ]$ and latitude $[85^\circ, 90^\circ]$; and Domain C: longitude $[-180^\circ, 180^\circ]$ and latitude $[-85^\circ, 85^\circ]$. The former two domains A and B are similar and the corresponding zone can be easily meshed, but the latter domain C are much more complicated due to the discontinuity from all the plate boundaries. Due to shortage of the dip angle data of the plate boundaries, we take all the faults are straight along the radial direction of the earth. The following procedure is then applied to generate the hexahedral shaped mesh of the whole model including the above plate boundaries: (1) The geometrical domain C is decomposed to several meshable sub-domains with the constraints of the plate boundaries as shown in Figs. 10 and 11a; (2) It is further grouped into different groups, and each group is meshed to the quadrilateral shaped mesh using paving/mapped mesh method and then transformed to an independent location for easy depiction, see Fig. 11b, c, d, e and f; (3) The above quadrilateral shaped mesh in Fig. 11b, c, d, e and f are respectively used as seed (fronts) to advance inward for the hexahedral shaped mesh generation with the sweeping method and then projected back to the 3D spherical space from the above projected plane coordinate system, as correspondingly shown in Fig. 12a, b, c, d and e. Here the crustal thickness is taken as 60 km; (4) Once the domain C is meshed, the blocks relevant with domains A and B can be easily meshed with the mapped method as described above but with the compatibility to the existing mesh/mesh seed along the common surfaces/edges shared with domain A and B, as denoted by A in Fig. 13a and b; (5) The hexahedral shaped meshes generated for the different groups/domains are assembled together with “welding” operation (i.e. node equivalent) except the nodes along the current plate boundaries and potential faults (that is treated as sticking frictional state when simulated using our finite element code (Xing et al. 2007a)), Figs. 13a, b and c. The generated meshes are optimized, and then the fault information could be extracted as shown in Fig. 14.

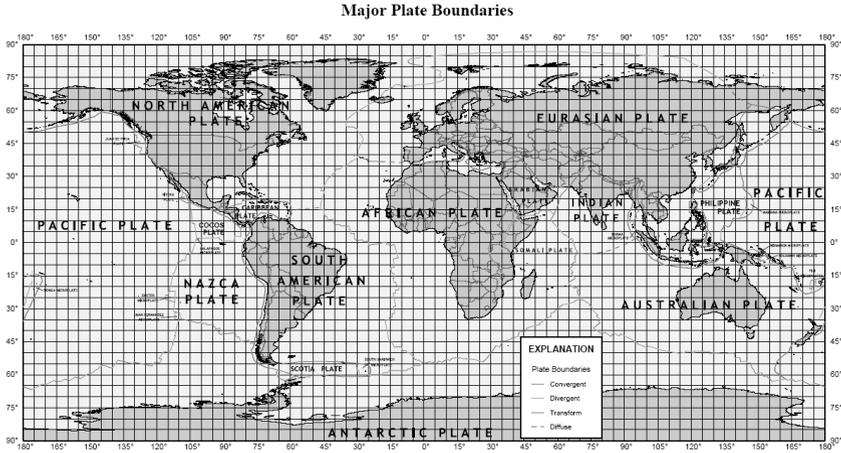


Fig. 10 The major plate boundaries of the entire Earth to be analyzed (courtesy of the U.S. Geological Survey)

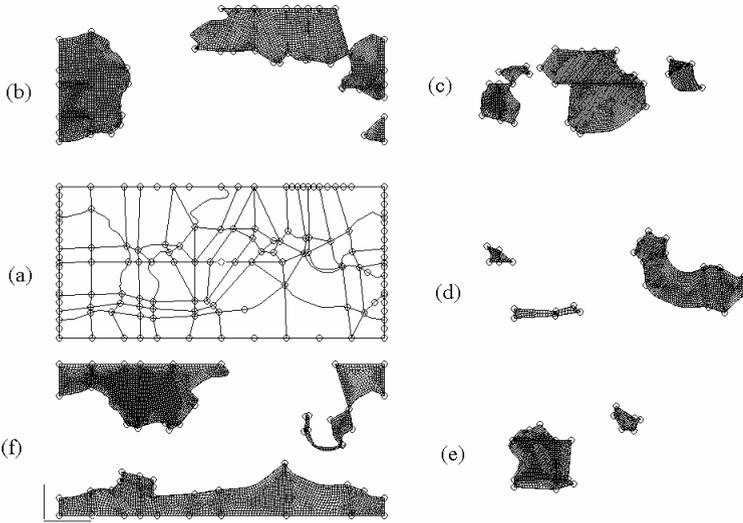


Fig. 11 The whole earth is divided into 3 domains A, B and C (a) The geometrical domain C here is decomposed to several meshable sub-domains (groups) with the constraints of the plate boundaries; (b, c, d, e and f) The quadrilateral shaped mesh generated with each group within the geometrical domain C (transformation is applied to each group in the figures for easy depiction)

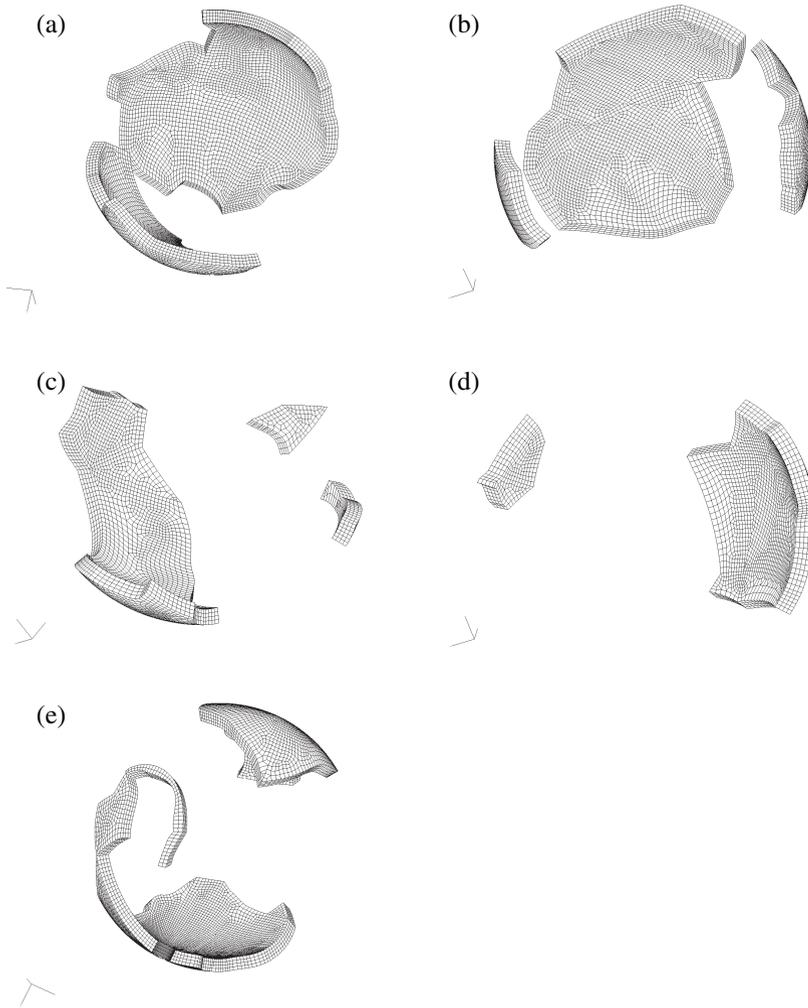


Fig. 12 (a, b, c, d and e) The hexahedral shaped mesh is generated using the sweeping method with the above quadrilateral shaped mesh respectively in (b, c, d, e and f) as seed (*fronts*) and then transferred to the 3D XYZ space from the above projected longitude-latitude plane

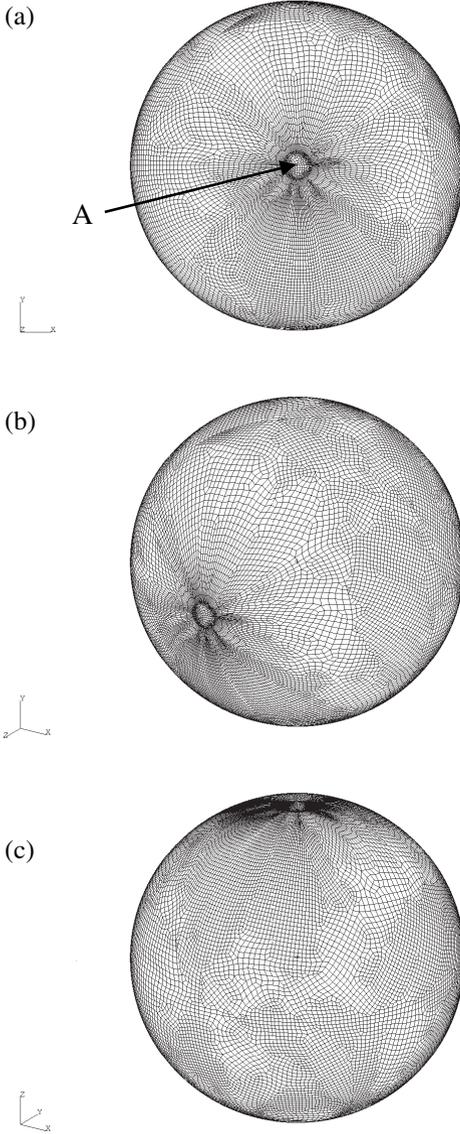


Fig. 13 The hexahedral shaped mesh generated after assembling all the components together in the whole domain (A, B and C) and viewed in the following coordinate systems (a) XY plane, (b) ZXY space and (c) XYZ space. A in (a) and (b) shows the meshes generated in the subdomain A or B after assembling with the meshes generated in the domain C

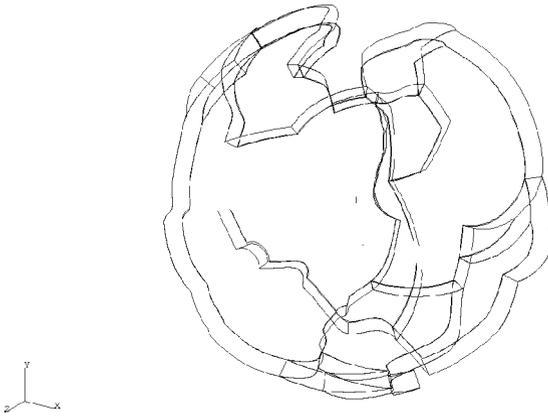


Fig. 14 The geometry and shape of the discontinuous surfaces (including the plate boundaries) in the hexahedral shaped mesh generated

4 Tetrahedral Mesh Generation

4.1 Introduction

Besides the hexahedral mesh as above, the tetrahedron is also very popular for mesh generation in 3D. All the above methods for all-hexahedral mesh generation could be directly used for the tetrahedral mesh generation, because a hexahedral mesh can be easily divided into tetrahedral meshes. The great advantage of tetrahedral shaped mesh generation over the above hexahedron is that it could be much easier to be automatically generated for a general physical domain using the Delaunay algorithm and advancing front technique et al. (see Table 1). The Delaunay algorithm (Delaunay 1934) is most widely used and has two following important properties: (1) the empty circle (or empty sphere) criterion, which means no node should be contained in the circumcircle (or circumsphere) of any triangle (or tetrahedral) element; (2) the maximum–minimum angle criterion, which means the smallest angle in the Delaunay triangular mesh is the largest one in all possible triangular mesh of a given node set. Such criteria were used and extended in mesh generation by Bowyer-Watson (Watson 1981), Baker (1989), Weatherill (1994) and George (1991) et al. For more details, please refer to the related surveys as above.

Tetrahedral shaped mesh generation has a wide variety of geological applications for accurate representation of complex geological structure and stratigraphy for the further numerical modeling of such as groundwater resource development, gas/oil reservoir system, waste disposal in a geologic repository. Besides those listed above, LaGriT is such a special-purposed software tool for importing and automatically producing unstructured tetrahedral mesh tuned to the special needs of geological and geoenvironmental applications developed at Los Alamos national laboratory (see <http://meshing.lanl.gov/>). Here, we focus on meshing a geological domain based on the available point data set with the regular or irregular meshes using our mesh generation code.

4.2 Automatic Tetrahedral Mesh Generation for the Stratigraphy Point Set

With the advanced measurement technique, the stratigraphy information in a certain geological domain is widely available, which may be described by a huge amount of point data. Therefore, the boundary/material surfaces can be extracted and defined by the relevant point data, which can be further described as triangular irregular networks (TINS). Figure 15a shows an example for a certain given domain to be analysed, which is composed of 3 different materials as denoted by different colours. The top surface and the setting of the different surfaces are shown in Fig. 15b and c, respectively. For a single stratigraphy, besides the above surfaces (i.e. the top and bottom surfaces), the other surfaces (the user defined normal regular plane to define the range of the above domain, i.e. the four vertical surfaces of this example) can be more easily described by triangular meshes which matching with the existing neighbour edges of the top and bottom surfaces), thus all these triangular surfaces could form a closed volume/representation of this single stratigraphy. And then a node distribution inside the closed volume domain is applied according to the prescribed characters, such as the material properties. Finally it is meshed into tetrahedral shaped meshes by using the Delaunay algorithm and further optimization. Here, assuming the material interfaces and the middle layer need to be specially addressed and thus the finer meshes generated around those areas correspondingly, Fig. 16a and b.

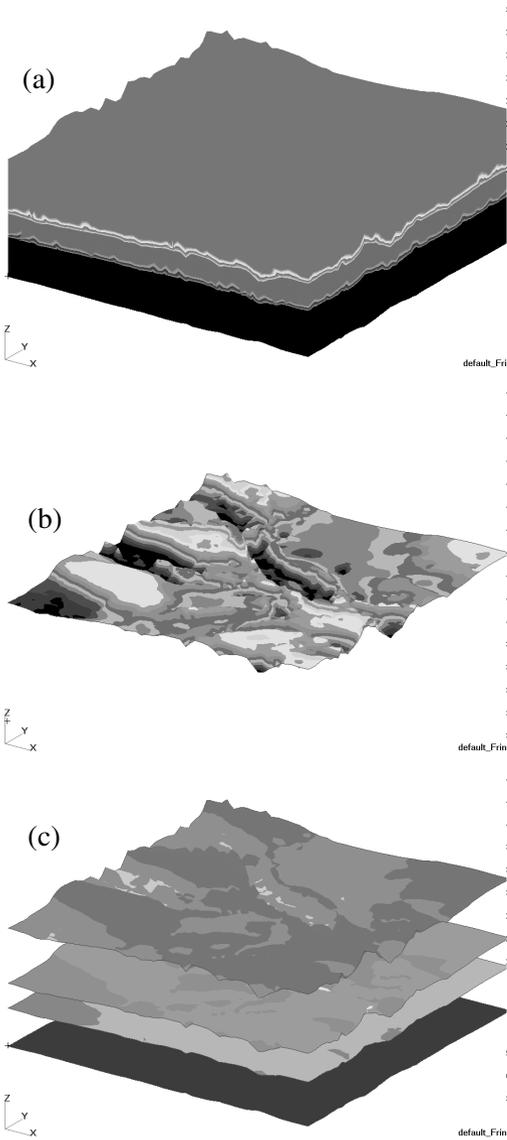


Fig. 15 Stratigraphy model for a certain given range of the region to be analysed (a) A solid model with 3 different materials as denoted using different colours; (b) the top surface and (c) the setting of the different surfaces of stratigraphy model

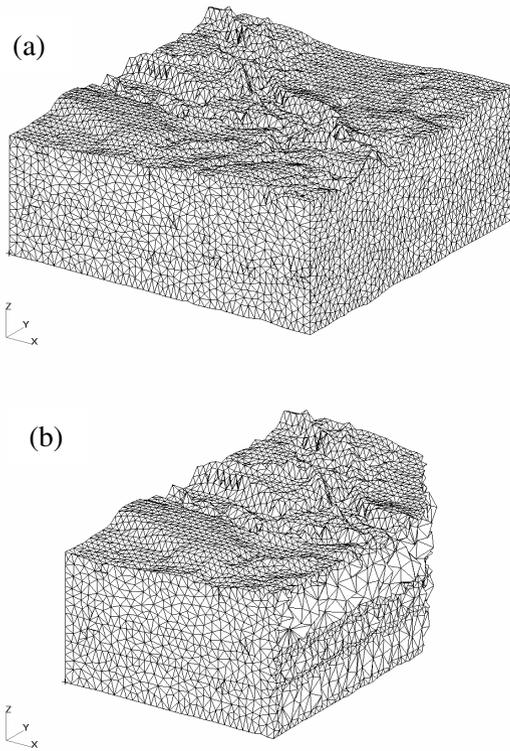
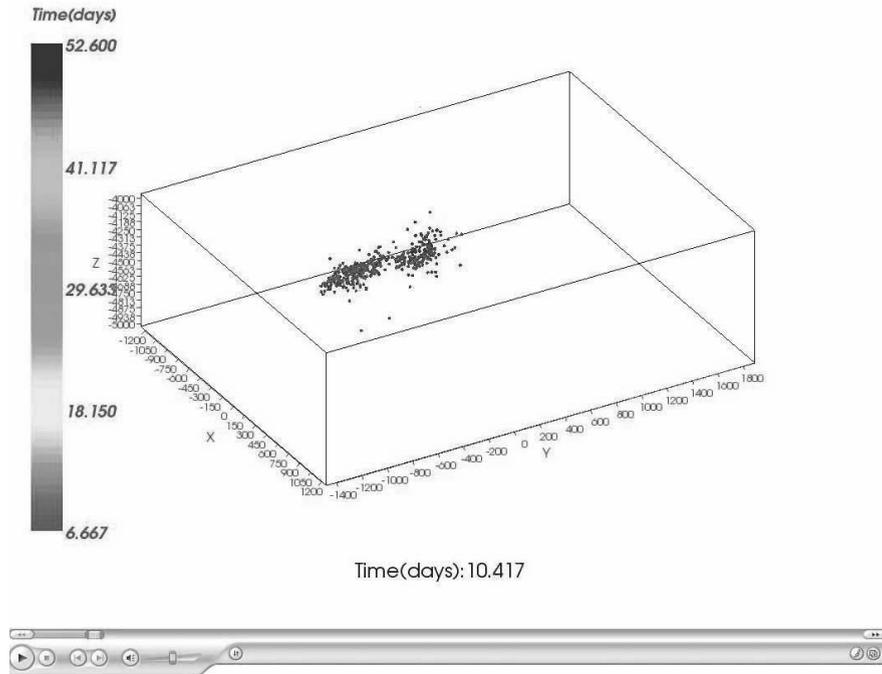


Fig. 16 The tetrahedral shaped mesh generated (a) in the whole domain and (b) the specified domain to show the internal mesh distribution

4.3 Visualizing and Meshing with the Microseismicity Data

Microseismicity is widely used in the mining industry including the hot dry/fractured rocks (HDR/HFR) geothermal exploitation to monitor and determine where and how the underground rupture proceeds during a certain processing, such as the widely applied hydraulic stimulation. The recorded microseismicity data provides the detailed location where an event (i.e. underground dynamic rupture) occurs at a certain time. During a hydraulic stimulation process, hundreds and thousands of microseismic events are recorded. With all the recorded data (i.e. an event location and its occurrence time), we take every event as an independent node/point

with its location in 3D space as recorded and colour it with its occurrence time, thus we can directly know where the underground dynamic rupture locates and how it proceeds with the time from the above information. Fig. 17a, b, c, d, e, f and g show an example within a certain range, which includes 11,724 events over 52 days recorded during the hydraulic stimulation process by the Geodynamics Limited (see Movie 1 for more details).



Movie 1 Visualization of the micro-seismicity proceeding with time in a hydraulic stimulation process (available on accompanying DVD)

Moreover, we want to generate the mesh using the recorded data for determining the solid domain of the ruptured zone and the further numerical analysis. The Delaunay algorithm may be the most suitable mesh generation method for such a point set and thus is applied here. However, the recorded data are not suitable to be directly used for mesh generation, because there may be lots of coincided points (including those

located too close) as well as some points which are far away from the main body data. A preprocess before mesh generation is taken as follows.

Define the recorded microseismicity data as the following scattered point set in 3D,

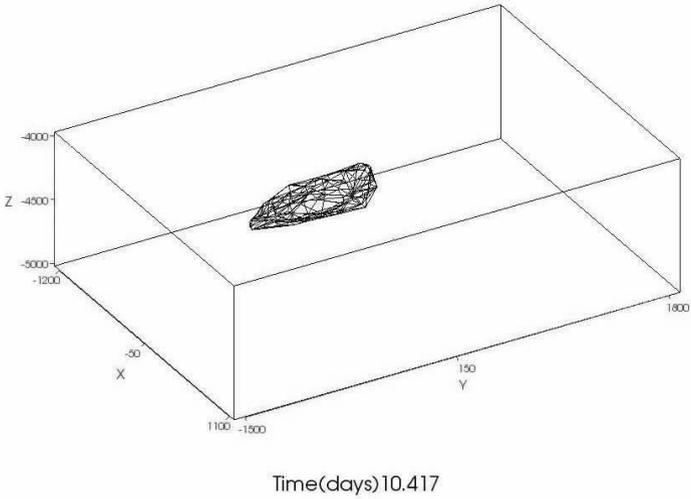
$$N = \{(x, y, z) \mid x, y, z \in \mathfrak{R}\} \quad (1)$$

Given a point $p_i(x_i, y_i, z_i) \in N$ and a tolerance $\varepsilon \in \mathfrak{R}$, the neighborhood point set $\delta(p_i, \varepsilon)$ is defined as follows:

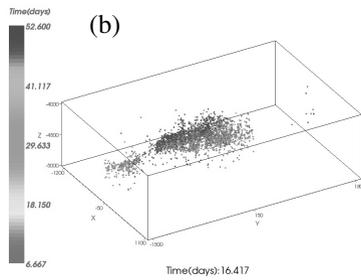
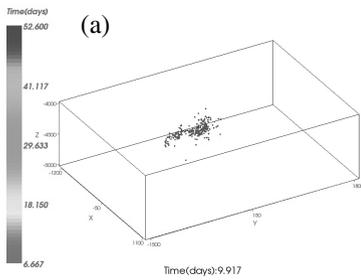
$$\delta(p_i, \varepsilon) = \{(x, y, z) \mid (x, y, z) \in N, \sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} < \varepsilon\} \quad (2)$$

Given the minimum and maximum tolerance, which are represented by ε_{\min} and ε_{\max} . For every point $p_i(x_i, y_i, z_i) \in N$, find the minimum neighborhood $\delta(p_i, \varepsilon_{\min})$ and the maximum neighborhood $\delta(p_i, \varepsilon_{\max})$. If $\delta(p_i, \varepsilon_{\min})$ is not empty, all other points except $p_i(x_i, y_i, z_i)$ in $\delta(p_i, \varepsilon_{\min})$ will be deleted. Given an integer number $\lambda \in I^+$, if $|\delta(p_i, \varepsilon_{\max})| < \lambda$, $p_i(x_i, y_i, z_i)$ will be deleted.

Once the above preprocessing procedure is finished, the Delaunay algorithm is used to form the convex hull using these point data, and the outside surface of the convex hull is then extracted. Furthermore, a certain internal node distribution (such as the variable mesh size control according to the outside surface) is determined and carried out. Finally, the Delaunay algorithm is applied to generate the tetrahedral shaped meshes. Figure 18a, b, c and d are the generated meshes with the above data at the different hydraulic stimulation stages (see Movie 2 for more details), which clearly show the solid domain which the ruptured zone roughly occupies with the microseismicity rupture proceeding and can also be used for the further finite element analysis. Here, the related parameters are set as: $\varepsilon_{\min} = 10$, $\varepsilon_{\max} = 100$, $\lambda = \log(|N|)$.



Movie 2 Visualization of the solid domain of the micro-seismicity rupture zone proceeding with time in a hydraulic stimulation process (available on accompanying DVD)



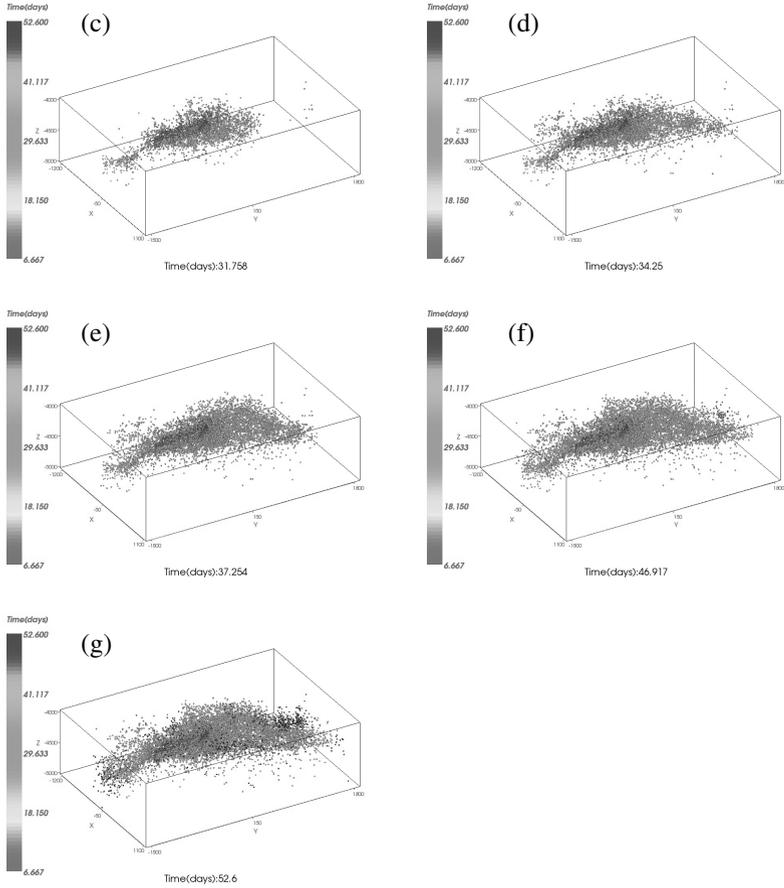
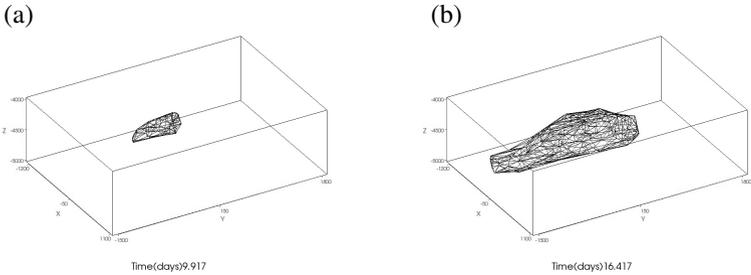


Fig. 17 Visualization of the micro-seismicity data in a hydraulic stimulation process at the different time (*days*): (a) 9.917, (b) 16.417, (c) 31.758, (d) 34.25, (e) 37.254, (f) 46.917 and (g) 52.60



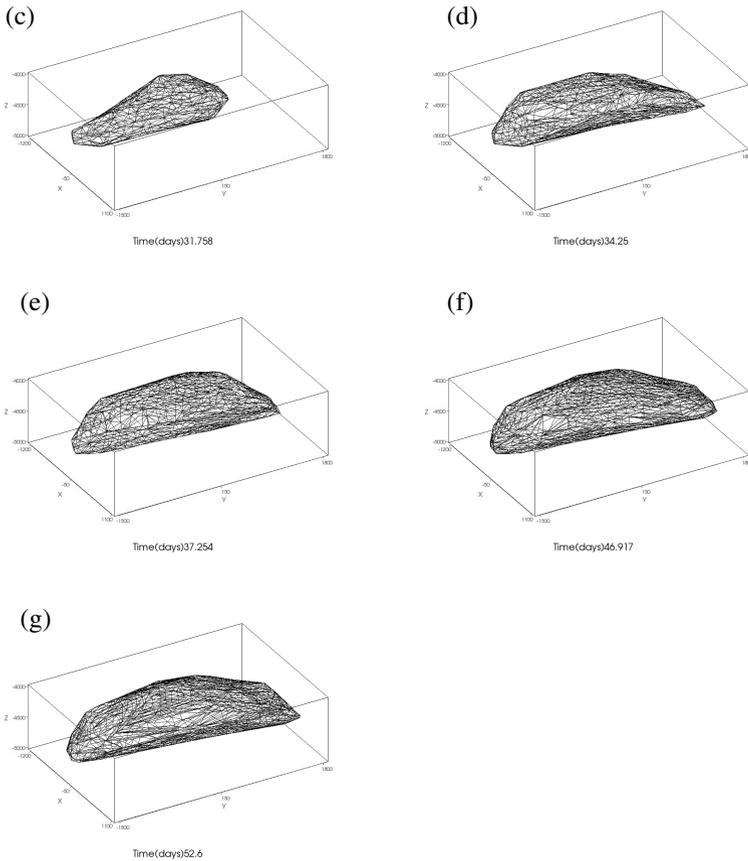


Fig. 18 Visualization of the solid domain of the rupture zone occupied in a hydraulic stimulation process at the different time (*days*): (a) 9.917, (b) 16.417, (c) 31.758, (d) 34.25, (e) 37.254, (f) 46.917 and (g) 52.60

Furthermore, the whole domain shown in Figs. 17 and 18 is chosen as the range to be analyzed. Following the similar procedure as above, the tetrahedral meshes generated using Delaunay algorithm are shown in Fig. 19a, b and c. Here, the mesh size on the 6 outer surfaces is taken as the same, but the nodal positions inside (i.e. mesh size) are controlled by the above microseismic data. Therefore, the inside mesh is quite heterogeneously distributed and its mesh size appears much smaller around the microseismicity concentrated zone, see Figs. 19 b and c.

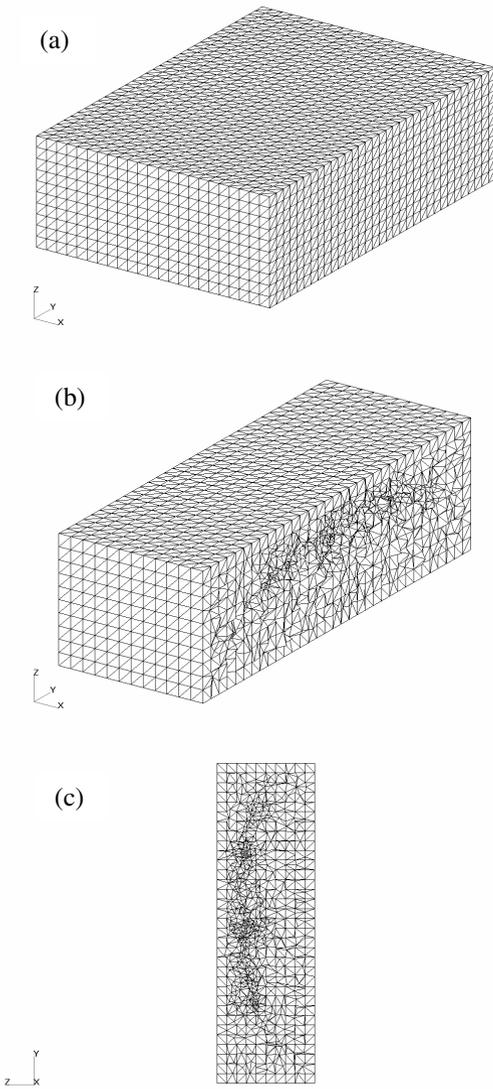


Fig. 19 The tetrahedral shaped mesh generated (a) in the full domain; (b) and (c) the internal mesh distribution seeing through a cross-section depicted in the XYZ space and the YZ plane respectively

5 Conclusions

Mesh generation is widely and successfully applied in the engineering computing, but it is relatively “new” for the geoscience community. This paper briefly introduces the relevant progresses and then discusses the possibility for applying and extending the above successful stories in engineering computing to geo-computing. For geoscience, the available input data are normally a large quantity of point data in the 3D space rather than the defined shapes and dimensions with reasonable tolerances provided by the industrial designers, thus quite different from the engineering cases. To deal with such geoscience data, this paper focuses on applying and/or extending the related methods to generate all hexahedral or tetrahedral shaped mesh in 3D for the different practical geoscience application examples based on the relevant progresses on geometrical modeling and hexahedral/tetrahedral shaped mesh generation. That includes all-hexahedral shaped mesh generation for a fracture dominated reservoir system, the South Australia interacting fault system and the entire earth models with the simplified/practical plate boundaries, and all-tetrahedral shaped mesh generation for a multi-layer underground geological model, and meshing with the microseismicity data recorded during a hydraulic stimulation process in a geothermal reservoir. It lays the foundation for the future research on such as the adaptive meshing and remeshing, parallel mesh generation as required from various different application cases.

Acknowledgements Support is gratefully acknowledged by the Australian Research Council Linkage project LP05609326 and Discovery project DP0666203, AuScope, the Queensland State Government, The University of Queensland, and SGI. The authors are grateful to Dr Doone Wyborn from the Geodynamics Limited, Professor Mike Sandiford from Melbourne University and Professor John Rundle from University of California at Davis for providing the related data used in this paper.

References

Baker TJ (1989) Automatic Mesh Generation for Complex Three-Dimensional Regions Using a Constrained Delaunay Triangulation. *Engineering with Computers*, Vol 5 pp 161–175

- Blacker T and Stephenson MB (1991) Paving: A New Approach to Automated Quadrilateral Mesh Generation. *International Journal for Numerical Methods in Engineering*, Vol 32 pp 811–847
- Borouchaki H and Frey PJ (1998) Adaptive Trangular-Quadrilateral Mesh Generation. *International Journal for Numerical Methods in Engineering*, Vol 41 pp 915–934
- Borouchaki H and George PL (1997) Aspects of 2-D Delaunay Mesh Generation. *International Journal for Numerical Methods in Engineering*, Vol 40 pp 1957–1975
- Borouchaki H, George PL and Lo SH (1996) Optimal Delaunay Point Insertion. *International Journal for Numerical Methods in Engineering*, Vol 39 pp 3407–3437
- Borouchaki H, Laug P and George PL (2000) Parametric Surface Meshing Using a Combined Advancing-Front Generalized Delaunay Approach. *International Journal for Numerical Methods in Engineering*, Vol 49 pp 233–259
- Borouchaki H and Lo SH (1995) Fast Delaunay Triangulation in Three Dimensions. *Computer Methods in Applied Mechanics and Engineering*, Vol 128 pp 153–167
- Calvo NA and Idelsohn SR (2000) All-Hexahedral Element Meshing: Generation of the Dual Mesh by Recurrent Subdivision. *Computer Methods in Applied Mechanics and Engineering*, Vol 182 pp 371–378
- Cheng GD and Li H (1996) New Method for Graded Mesh Generation of Quadrilateral Finite Elements. *Computers and Structures*, Vol 59 Num 5 pp 823–829
- Chiba N, Nishigaki I, Yamashita Y, Takizawa C and Fujishiro K (1998) A Flexible Automatic Hexahedral Mesh Generation by Boundary-Fit Method. *Computer Methods in Applied Mechanics and Engineering*, Vol 161 pp 145–154
- Delaunay BN (1934) Sur la Sphere. *Vide. Izvestia Akademia Nauk SSSR*, VII Seria, Otdelenie Matematicheskii i Estestvennyka Nauk, Vol 7 pp 793–800
- Dewhurst D, Vangavolu S, Wattrick H (1995) The Combination of Hexahedral and Tetrahedral Meshing Algorithms. *Proceeding 4th International Meshing Roundtable* pp 291–304
- Du Q and Wang DS (2003) Tetrahedral Mesh Generation and Optimization Based on Centroidal Voronoi Tessellations. *International Journal for Numerical Methods in Engineering*, John Wiley & Sons, Ltd., Vol 56 Num 9 pp 1355–1373
- Dziewonski AD and Anderson DL (1981) Preliminary Reference Earth Model. *Physics of the Earth and Planetary Interiors*, Vol 25 Num 2 pp 97–356
- Frey P, Benoit S and Gautherie M (1994) Fully Automatic Mesh Generation for 3-D Domains Based Upon Voxel Sets. *International Journal for Numerical Methods in Engineering*, John Wiley, Num 37 pp 2735–2753
- George PL (1997) Improvements on Delaunay-Based Three-Dimensional Automatic Mesh Generator. *Finite Elements in Analysis and Design*, Elsevier, Vol 25 pp 297–317
- George PL, Hecht F and Saltel E (1991) Automatic Mesh Generator with Specified Boundary. *Computer Methods in Applied Mechanics and Engineering*, North-Holland, Vol 92 pp 269–288

- George PL and Seveno E (1994) The Advancing-Front Mesh Generation Method Revisited. *International Journal for Numerical Methods in Engineering*, Wiley, Vol 37 pp 3605–3619
- Hariya M, Nishigaki I, Kataoka I, Hiro Y (2006) Automatic Hexahedral Mesh Generation with Feature Line Extraction. *Proceeding 15th Round Table Meshing* pp 453–468
- Joe B (1991a) GEOMPACK – A Software Package for the Generation of Meshes Using Geometric Algorithms. *Advances in Engineering Software*, Elsevier, Vol 13 Num 5 pp 325–331
- Joe B (1991b) Construction of Three-Dimensional Delaunay Triangulations Using Local Transformations. *Computer Aided Geometric Design*, Elsevier Science Publishers (North-Holland), Num 8 pp 123–142
- Joe B (1991c) Delaunay Versus Max–Min Solid Angle Triangulations For Three-Dimensional Mesh Generation. *International Journal for Numerical Methods in Engineering*, John Wiley & Sons, Vol 31 pp 987–997
- Joe B (1995) Construction of Three-Dimensional Improved-Quality Triangulations Using Local Transformations. *SIAM Journal of Science Computing*, Vol 16 pp 1292–1307
- Kageyama A and Sato T (2004) The ‘Yin-Yang Grid’: An Overset Grid in Spherical Geometry. *Geochem. Geophys. Geosyst.*, Q09005, doi:10.1029/2004GC000734
- Kanai T, Makinouchi A, Oishi Y (2000) Development of Tectonic CAD/Database Systems. In *Abstracts of International Workshop on Solid Earth Simulation and ACES WG Meeting* (Ed. Mitsu’ura M et al.). Tokyo, Jan 17–21
- Knupp PM (1998) Next-Generation Sweep Tool: A Method For Generating All-Hex Meshes on Two-And-One-Half Dimensional Geomtries. *Proceedings the 7th International Meshing Roundtable* pp 505–513
- Knupp PM (1999) Applications of Mesh Smoothing: Copy, Morph, and Sweep on Unstructured Quadrilateral Meshes. *International Journal for Numerical Methods in Engineering*, Wiley, Vol 45 pp 37–45
- Lai MW, Benzley S and White D (2000) Automated Hexahedral Mesh Generation by Generalized Multiple Source to Multiple Target Sweeping. *International Journal for Numerical Methods in Engineering*, John Wiley, Vol 49 Num 1 pp 261–275
- Lau TS and Lo SH (1996) Finite Element Mesh Generation Over Analytical Surfaces. *Computers and Structures*, Elsevier Science Ltd., Vol 59 Num 2 pp 301–309
- Lee CK (2003) Automatic Metric 3D Surface Mesh Generation Using Subdivision Surface Geometrical Model. Part 2: Mesh Generation Algorithm and Examples. *International Journal for Numerical Methods in Engineering*, John Wiley & Sons, Ltd., Vol 56 Num 11 pp 1615–1646
- Lee CK and Hobbs RE (1999) Automatic Adaptive Finite Element Mesh Generation Over Arbitrary Two-Dimensional Domain Using Advancing Front Technique. *Computers and Structures Pergammon*, Vol 71 pp 9–34
- Lee YK and Lee CK (2002) Automatic Generation of Anisotropic Quadrilateral Meshes on Three-Dimensional Surfaces Using Metric Specifications. *International Journal for Numerical Methods in Engineering*, John Wiley & Sons, Ltd., Vol 53 Num 12 pp 2673–2700

- Lee YK, Lee CK (2003) A New Indirect Anisotropic Quadrilateral Mesh Generation Scheme with Enhanced Local Mesh Smoothing Procedures. *International Journal for Numerical Methods in Engineering*, John Wiley & Sons, Ltd., Vol 58 Num 2 pp 277–300
- Lee CK and Lo SH (1994) A New Scheme for the Generation of a Graded Quadrilateral Mesh. *Computers and Structures Pergamon* Vol 52 Num 5 pp 847–857
- Lee DT and Schacter BJ (1980) Two Algorithms for Constructing a Delaunay Triangulation. *International Journal of Computer and Information Sciences*, Vol 3 Num 9 pp 219–242
- Leland RW, Melander D, Meyers R, Mitchell S, Tautges T (1998) The Geode Algorithm: Combining Hex/Tet Plastering, Dicing and Transition Elements for Automatic, All-Hex Mesh Generation. *Proceeding 7th International Meshing Roundtable* pp 515–521
- Lo SH (1991a) Volume Discretization into Tetrahedra-I. Verification and Orientation of Boundary Surfaces. *Computers and Structures*, Pergamon Press, Vol 39 Num 5 pp 493–500
- Lo SH (1991b) Volume Discretization into Tetrahedra – II. 3D Triangulation by Advancing Front Approach. *Computers and Structures Pergamon*, Vol 39 Num 5 pp 501–511
- Lo SH (1992) Generation of High Quality Gradation Finite Element Mesh. *Engineering Fracture Mechanics*, Pergamon, Vol 2 Num 41 pp 191–202
- Lohner R (1996a) Progress in Grid Generation via the Advancing Front Technique. *Engineering with Computers*, Springer-Verlag, Vol 12 pp 186–210
- Lohner R (1996b) Extensions and Improvements of the Advancing Front Grid Generation Technique. *Communications in Numerical Methods in Engineering*, John Wiley & Sons, Ltd., Vol 12 pp 683–702
- Lohner R (1996c) Regridding Surface Triangulations. *Journal of Computational Physics*, Academic Press, Vol 126 pp 1–10
- Lohner R and Cebal JR (2000) Generation of Non-Isotropic Unstructured Grids via Directional Enrichment. *International Journal for Numerical Methods in Engineering*, John Wiley, Vol 49 Num 1 pp 219–232
- Lohner R and Eugenio O (1998) An Advancing Front Point Generation Technique. *Communications in Numerical Methods in Engineering*, Wiley, Vol 14 pp 1097–1108
- Loic M (2001) A New Approach to Octree-Based Hexahedral Meshing. *Proceedings 10th International Meshing Roundtable*, pp 209–221
- Owen SJ, Canann S, Saigal S (1997) Pyramid Elements for Maintaining Tetrahedra to Hexahedra Conformability. *Trends in Unstructured Mesh Generation AMD*, Vol 220 pp 123–129
- Owen SJ, Saigal S (2000) H-Morph: An Indirect Approach to Advancing Front Hex Meshing. *International Journal for Numerical Methods in Engineering*, John Wiley & Sons, Ltd., Vol 49 Num 1–2 pp 289–312
- Owen SJ, Staten ML, Canann SA and Saigal S (1999) Q-Morph: An Indirect Approach to Advancing Front Quad Meshing. *International Journal for Numerical Methods in Engineering*, Wiley, Vol 9 Num 44 pp 1317–1340

- Ray M, Tautges T, Tuchinsky P (1998) The “Hex-Tet” Hex-Dominant Meshing Algorithm as Implemented in CUBIT. Proceeding 7th International Meshing Roundtable pp 151–158
- Rosenbaum G, Lister GS and Duboz C (2002) Reconstruction of the tectonic evolution of the western Mediterranean since the Oligocene. In: Rosenbaum, G. and Lister, G. S. 2002. Reconstruction of the evolution of the Alpine-Himalayan Orogen. *Journal of the Virtual Explorer*, Vol 8 pp 107–126
- Schneiders R (1996) A Grid-Based Algorithm for the Generation of Hexahedral Element Meshes. *Engineering with Computers*, Vol 12 pp 168–177
- Schneiders R (1997) An Algorithm for the Generation of Hexahedral Element Meshes Based on an Octree Technique. Proceedings 6th International Meshing Roundtable pp 183–194
- Schneiders R and Bunten R (1995) Automatic Generation of Hexahedral Finite Element Meshes. *Computer Aided Geometric Design*, Elsevier, Vol 12 pp 693–707
- Schneiders R, Schindler R and Weiler F (1996) Octree-Based Generation of Hexahedral Element Meshes. Proceedings 5th International Meshing Roundtable pp 205–216
- Scott MA, Earp MN, Benzley SE and Stephenson MB (2005) Adaptive Sweeping Techniques. Proceedings the 14th International Meshing Roundtable pp 417–432
- Shephard MS and Marcel KG (1991) Automatic Three-Dimensional Mesh Generation by the Finite Octree Technique. *International Journal for Numerical Methods in Engineering*, Wiley, Vol 32 pp 709–749
- Shephard MS and Marcel KG (1992) Reliability of Automatic 3D Mesh Generation. *Computer Methods in Applied Mechanics and Engineering*, North-Holland, Vol 101 pp 443–462
- Shepherd J, Mitchell SA, Knupp P, and White D (2000) Methods for Multisweep Automation. Proceedings of the 9th International Meshing Roundtable, Sandia National Laboratories, pp. 77–87
- Shewchuk JR (2002) Delaunay Refinement Algorithms for Triangular Mesh Generation, *Computational Geometry: Theory and Applications*, Vol 22 Num 1–3 pp 21–74, May 2002
- Shin BY and Sakurai H (1996) Automated Hexahedral Mesh Generation by Swept Volume Decomposition and Recomposition. Proceeding 5th International Meshing Roundtable pp 273–280
- Staten ML, Canann SA and Owen SJ (1998) BMSWEEP: Locating Interior Nodes During Sweeping. Proceeding the 7th International Meshing Roundtable pp 7–18
- Staten ML, Owen SJ, Blacker TD (2005) Unconstrained Paving & Plastering: A New Idea for all Hexahedral Mesh Generation. Proceedings the 14th International Meshing Roundtable pp 399–416
- Taghavi R (2000) Automatic Block Decomposition Using Fuzzy Logic Analysis. Proceeding 9th International Meshing Roundtable pp 187–192
- Takahashi H and Shimizu H (1991) A General Purpose Automatic Mesh Generation Using Shape Recognition Technique. *Compt. Engrg. ASME*, Vol 1 pp 519–526

- Tu T and O'Hallaron DR (2004) Extracting Hexahedral Mesh Structures from Balanced Linear Octrees. Proceedings 13th International Meshing Roundtable Williamsburg VA pp 191–200
- Watson DF (1981) Computing the Delaunay Tesselation with Application to Voronoi Polytopes. *The Computer Journal*, Vol 24 Num 2 pp 167–172
- Weatherill NP and Hassan O (1994) Efficient Three-dimensional Delaunay Triangulation with Automatic Point Creation and Imposed Boundary Constraints. *International Journal for Numerical Methods in Engineering*, Wiley, Num 37 pp 2005–2039
- White D, Saigal S and Owen S (2004) CCSweep: Automatic Decomposition of Multi-Sweep Volumes. *Engineering with Computers*, Vol 20 pp 222–236
- White D, Tautges T and Timothy J (2000) Automatic Scheme Selection for Toolkit Hex Meshing. *International Journal for Numerical Methods in Engineering*, Vol 49 pp 127–144
- Wright JP and Alan GJ (1994) Aspects of Three-Dimensional Constrained Delaunay Meshing. *International Journal for Numerical Methods in Engineering*, Wiley, Num 37 pp 1841–1861
- Xing HL, Miyamura T, Makinouchi A, Homma T, Kanai T, Oishi Y (2001) Development of High Performance Finite Element Software System for Simulation of Earthquake Nucleation and Development, *Exploration Geodynamics*. (Eds. Moresi L, Muller D and Hobbs B). Western Australia, pp 178–188
- Xing HL, Makinouchi A and Mora P (2007a) Finite Element Modeling of Interacting Fault System, *Physics of the Earth and Planetary Interiors*, Vol 163 pp 106–121. DOI 10.1016/j.pepi.2007.05.006
- Xing HL and Mora P (2003) Mesh Generation Software Survey, Technical Report of ACcESS (ESSCC, The University of Queensland). pp 1–20
- Xing HL and Mora P (2006) Construction of an Intraplate Fault System Model of South Australia, and Simulation Tool for the iSERVO Institute Seed Project. *Pure and Applied Geophysics*, Vol 163 pp 2297–2316. DOI 10.1007/s00024-006-0127-x
- Xing HL, Zhang J and Yin C (2007b) A Finite Element Analysis of Tidal Deformation of the Entire Earth with a Discontinuous Outer Layer. *Geophysical Journal International*, Vol 170 Num 3 pp 961–970. DOI 10.1111/j.1365-246X.2007.03442.x
- Yamakawa S and Shimada K (2003) Anisotropic Tetrahedral Meshing via Bubble Packing and Advancing Front. *International Journal for Numerical Methods in Engineering*, John Wiley & Sons, Ltd., Vol 57 Num 13 pp 1923–1942
- Yerry MA and Shephard MS (1984) Automatic Three-Dimensional Mesh Generation by the Modified Octree Technique. *International Journal for Numerical Methods in Engineering*, John Wiley, Num 20 pp 1965–1990