Computational Geosciences
https://doi.org/10.1007/s10596-024-10290-1

ORIGINAL PAPER

Automation of the meshing process of geological data

Sui Bun Lo[1] · Oubay Hassan[1] · Jason Jones[1] · Xiaolong Liu[2] · Nevan C Himmelberg[2] · Dean Thornton[2]

Received: 3 October 2023 / Accepted: 11 April 2024

Abstract

This work proposes a novel meshing technique that is able to extract surfaces from processed seismic data and integrate surfaces that were constructed using other extraction techniques. Contrary to other existing methods, the process is fully automated and does not require any user intervention. The proposed system includes an approach for closing the gaps that arise from the different techniques used for surface extraction. The developed process is able to handle non-manifold domains that result from multiple surface intersections. Surface and volume meshing that comply with user specified mesh control techniques are implemented to ensure the desired mesh quality. The integrated procedures provide a unique facility to handle geotechnical models and accelerate the generation of quality meshes for geophysics modelling. The developed procedure enables the creation of meshes for complex reservoir models to be reduced from weeks to a few hours. Various industrial examples are shown to demonstrate the practicable use of the developed approach to handle real life data.

Keywords Reservoir modelling · Surface mesh optimization · Volume mesh generation · Surface intersection · Surface extension

Mathematics Subject Classification (2010) 65M50

1 Introduction

The last few decades have seen rapid advances in computational techniques that have drastically changed the engineering design process. Reservoir modelling has become a critical tool that needs to be used by the hydrocarbon industry to maximise the yield of new and existing oil fields. Aggressive use of computational methods has regularly been used alongside field testing to enhance production. The process starts by transforming seismic and bore drilling data into a geological model describing the structure of the reservoir. This model allows simulations to be performed which give insight to make production and investment choices. Considering the scale of these models, that are hundreds or thousands of meters in each direction, and the relatively small magnitude of the displacements, centimetres to a few meters at most, that will be induced by production activities over the lifetime of the reservoir, the entire structure of a reservoir is considered static [1], hence, a well constructed model will be invaluable to the company over the entire production period[2].

Studying the geo-mechanical behaviour of reservoirs has gained importance in the petroleum industry. Changes in pore pressure that result from depletion or injection lead to changes in effective stress. The changes of the effective stress may result in compaction of the reservoir which is advantageous for poorly compacted reservoirs. However, it may cause surface subsidence and create damage to well equipment. In addition, in the case of fractured and faulted reservoirs, the stress changes affect the fracture conductivity.

In order to properly assess the implication of the change in the effective stress that result from depletion or injection, an accurate model to capture the large scale geologic structures of the region of interest has to be constructed. This should include the identification of the surfaces that separate the various layers forming the reservoir together with the faults that have formed within the region of interest. However, when dealing with large scale geologic structures, it is not required that the interface surface conform to a high-resolution rep-

✉ Sui Bun Lo
s.lo@swansea.ac.uk

[1] Zienkiewicz Centre for Computational Engineering, Faculty of Science and Engineering, Swansea University, Fabian Way, SA1 8EN Swansea, Wales, United Kingdom

[2] Subsurface Department, Chevron Corporation, 1500 Louisiana, 77002 Houston, Texas, United States of America

Published online: 07 May 2024

resentation of stratigraphy as sometimes needed to capture details of fluid transport phenomena or to simulate near-well stability.

The creation process of reservoir models involves analysing seismic data and well-log data to identify the surfaces of the structures. Reservoirs are known to have very thin layers, with sharp edges, holes and small-scale tubular interconnections between layers. Extracting information from seismic data and well-log data is a complicated process. The state of the art techniques use manual or automated methods for seismic interpretation. However, regions with little seismic energy, particularly below and around salt or steep faults typically have less fidelity, thus the higher uncertainty. Other techniques such as ground penetrating radar, electrical resistivity tomography and ground magnetic surveys to image the subsurface are more effective for detecting faults and shallow geological structures

The interpreted seismic data is typically defined on very high resolution grids that normally have a very poor aspect ratio. When appropriate, seismic velocity cubes are used to extract high contrast objects. Even after significant image processing fine–scale features are widespread within the domain. Hence, the surface representation of geological interfaces regularly require user intervention. Furthermore, other geological features such as faults are extracted using other exploration techniques. These surfaces are then incorporated within the extracted domain prior to its use in downstream processes.

It is important for mesh extraction, in geotechnical modelling, to preserve and maintain the topology of the inner layers, eliminate topologically insignificant small features and produce valid, good quality meshes that have an optimum number of elements. Over the years various algorithms have been developed to transform seismic data into reservoir models [3–5]. Early numerical modelling processes utilised structured Cartesian grids, but this method could not represent complex geological features. As such, unstructured meshes have become popular for their ability to represent complex structures accurately. Caumon et al. [6], used manually defined boundary representations to create a sealed geological model; Jackson et al. [7] proposed a scheme for creating a tetrahedral mesh from surfaces of each layer of the reservoir. Zhang et al. [8] utilises NURBS and Coons patches to define the geometry of a reservoir and uses open source tools to create a volume mesh.

The various procedures to extract all the surfaces of interest are usually performed by different teams. The remit of these teams is to provide a triangulation that represents the surface of interest with often little attention paid to the validity of the produced surface representation. This results in inconsistencies in the input surface geometry such as the orientation, self intersection and gaps that need to be addressed prior to, or during, the meshing process. The techniques that incorporate all the required surface produced need to ensure conformity and water tightness. Seo et al. [9] developed a surface mesh intersection algorithm to support automatic mesh merging; Khan et al. [10] handled poorly triangulated regions with a hole cutting and filling method. These algorithms rely on the intersection to occur; however in many instances these geological features may not intersect and the surfaces would require days of manual intervention to produce a valid mesh.

Once the surfaces of a reservoir are extracted, and the additional surfaces that are captured using other exploration techniques have been included within the domain of interest, mesh enhancement is normally employed. The aim is, first, to improve the quality of the surface mesh and second, to achieve a mesh resolution that is appropriate for the intended simulation. Surface remeshing is the most widely used technique for the generation of high-quality surface meshes from a given surface triangulation that lacks an underlying surface definition [11–15]. This procedure includes, edge splitting/collapsing together with edge swapping and nodal smoothing. For a successful implementation of the remeshing technique, ridges and corners have to be identified. However, the automation of this identification procedure is a non-trivial task and may result in a degradation of the surface conformity, whilst manual intervention results in a degradation of the efficiency and robustness of the procedure.

In this paper, we aim to devise an integrated process, that starts from surfaces extracted using different interpretation and modelling techniques to represent salt bodies, fault surfaces, and stratigraphic horizons, and generate surface and volume meshes that exhibit the quality required for the targeted finite element simulation. The process will consider the inconsistencies that arise from the different data interpretation techniques used to extract the surfaces prior and during the meshing process. The process will allow the hydrocarbon industry to quickly create a mesh for simulation bypassing most of the manual intervention and reducing the meshing period for complex domains from days to hours. The developed approach can start by extracting the interface surface between two materials from provided data on regular intervals using a marching cube technique. The development allows for the insertion of triangulated surfaces that were extracted using different geological interpretation methods. A technique is then devised to ensure that gaps between stratigraphic horizons and faults, are appropriately closed. The resulting surfaces are then passed through a mesh enhancement procedure to ensure the elements comply with the user defined mesh spacing and are of the good quality for the intended geomechanics simulations that utilize finite-element methods. The process is then completed by the generation of a high quality volume mesh utilising the in-house FLITE tetrahedral mesh generation system. To our knowledge this integrated capability does not exist in any single software package and, hence, it provides a unique, integrated facility for the creation of computational models for reservoir modelling.

The remaining part of the paper is organised as follow. Section 2 describes the facility added for the extraction of the interface surface between two domains with user defined cube velocity. Section 3 provides the details of the surface combination procedure. Section 4 details the surface remeshing techniques that are used to enhance the surface triangulation of the domain. Section 5 describes the applications of the technique to three test cases.

2 Surface extraction from seismic data

The geological interpretation process starting from 3D seismic and well-log data is, in general, a complex process that incorporates manual and/or automated methods. The incorporation of these techniques is beyond the remit of this paper. In this work we utilise the wave velocity variations in the subsurface that are derived from 3D seismic data. This data is extracted at regular intervals in the horizontal and vertical directions throughout the region of interest and is used to identify the interface surface between two well defined materials.

The surface extraction starts from the provided velocity, the azimuth that measures the direction from the north and the dip that measures the angle at which a rock layer descends from the horizontal given at prescribed locations defined by their coordinates (x, y, z), where x and y are the coordinates on the horizontal surface, and z is the vertical depth below the surface. Traditionally, the locations are uniformly spaced in each direction, however, higher resolution is normally provided in the vertical direction.

The initial data provided can map into an anisotropic Cartesian grid forming cuboid cells. Each prescribed location of the provided data will be assumed to coincide with the centre of one cuboid cell and has a constant value within the cell. However, if desired, the developed technique enables the user to reduce the available data in each direction separately by merging multiple cuboid cells into one that has the averaged values of the merged cells. The merging will reduce the starting surface triangulation that is formed from the Cartesian grid which, in turn, will make the remeshing process faster. This is beneficial if coarser meshes are utilised at the initial stage of the exploration .

The marching cube algorithm, originally designed for medical imaging, has been shown to have the ability to automatically reconstruct a mesh from geological data [16]. The technique was also used in conjunction with level set [17] in order to extract an initial polygonal surface. A marching cube type algorithm is utilised to identify cells that have velocity within a user prescribed velocity range that correspond to certain strata. This process results in multiple closed regions formed from connected cubes sharing one or more faces. These regions could be disconnected or connected at corner points or edge of cubes belong to two separate regions. The implementation, enables the application of different levels of data reduction and smoothing to remove features that are insignificant for the intended modelling:

- remove regions that are formed by a single cell
- remove single cells that are connected to a region through a single face only. This process does not apply to a region that can be separated into two separate regions if a single face is ignored
- remove regions that have less than a user defined continuous number of cells in a given direction, i.e, discard the associated cubes from the selection
- separate regions that are joined by an edge.
- separate regions that are connected at corner points.

Two region separation techniques have been implemented. The first technique, duplicates the corner points or the common edges and their nodal points. For each region, the duplicated points are displaced, by a user specified value, in the direction of the inward normal. In the second technique, one of the cells that contain a corner points or a common edge is discarded from the selection. The second technique has been utilised for the example shown in this paper as it prevents the occurrence of surfaces with close proximities that would result in highly stretched volume meshes.

Following the above stage, the cells are divided into tetrahedral elements and their connectivities are used to determine internal and external boundaries. This results in a stair case boundary representation. The surface mesh obtained at this stage will be conforming and valid, however, its quality depends greatly on the degree of anisotropy inherited from the starting 3D anisotropic grid that usually has more resolution in the vertical direction than the horizontal directions, resulting in stretched cells. The dip and azimuth data are then utilised to smooth the edges of the stair case surface and align the surface triangulation with orientation of the interface layer. While this process does not improve the mesh quality of the triangulated stair-cased surface mesh, it reduces the discontinuities that are inherited in a surface triangulation from structured grid. The surface enhancement procedure described in Section 4 is then applied to the extracted surface mesh to ensure elements with adequate qualities that comply with the desired user specified spacing.

In the case that multiple strata are required to be extracted, the above procedure can be applied iteratively starting from a velocity range that encompasses all the strata of interest. A velocity sub–range is then introduced to sub-divide the extracted domains into two sub–domains. Further subdivision can then be carried out on each of these sub domains until all the strata of interest are captured. Each resulting stratum will consist of one or more enclosed regions.

3 Combine multiple surfaces

In most cases, the extracted geological surfaces needs to be bounded to enclose a specific region of interest. This process involves the insertion of planes that intersect the domain at the desired locations. In addition, it is established that in many cases, the seismic data is not able to capture all the geological features that are of interest. Surfaces that have been constructed using other means of exploration will have to be incorporated within the extracted domain. However, often these additional surfaces are not fully compatible with the underlying surface mesh of the domain. This can be seen in Fig. 2, where a gap between the reconstructed fault surface and the interface layer can be observed. Traditionally, manual intervention requiring hours, if not days, is needed to ensure conformity and construct a water tight domain. The proposed process detects the presence of gaps and perform local surface extensions that enable the closure of the unwanted gaps.

The efficient implementation of all the proposed procedures depend heavily on the adopted data structure. It is also dependent on the ability to identify ridges and open surfaces. The following definitions have been adopted in the proceeding sections:

- The domain or region of interest is defined by one or more input surfaces with the orientation specified such that the final volumetric mesh conforms to one side only.
- The domain of interest can also be constructed by truncating the geological surfaces. This is achieved by extruding a user defined polygon in the vertical direction to form a polyhedral domain with planar faces.
- Internal surfaces can be completely contained within the domain of interest or crossing the boundary of the domain. A conformal final volumetric mesh will exist on both sides of these surfaces.
- A ridge can either be determined using a user specified value of a dihedral angle between two adjacent triangles or determined from the intersection of two or more surfaces.

Figure 1 shows a schematic of a typical domain of interest with indication to the different type of entities that are used in the following sections.

3.1 Data structure

Unstructured mesh generation and mesh modification are dynamic processes that require continuous updating of the data structure. For an optimum implementation of the operations that are required in these processes, link lists and ADTs data structures have been adopted for efficient searching algorithms.

The data structure adopted, $\mathcal{D}$, contains various substructures to facilitate the continuous update of the mesh information;

- Element connectivity matrix, $\mathcal{C}^{\mathcal{D}}$, that contains the index of the three nodes of each triangular element and the surface number to which is the triangle belong.
- Edge connectivity matrix, $\mathcal{E}^{\mathcal{D}}$, that contains the index of the two nodes forming the edge and the number of the two adjacent triangles of the edge.

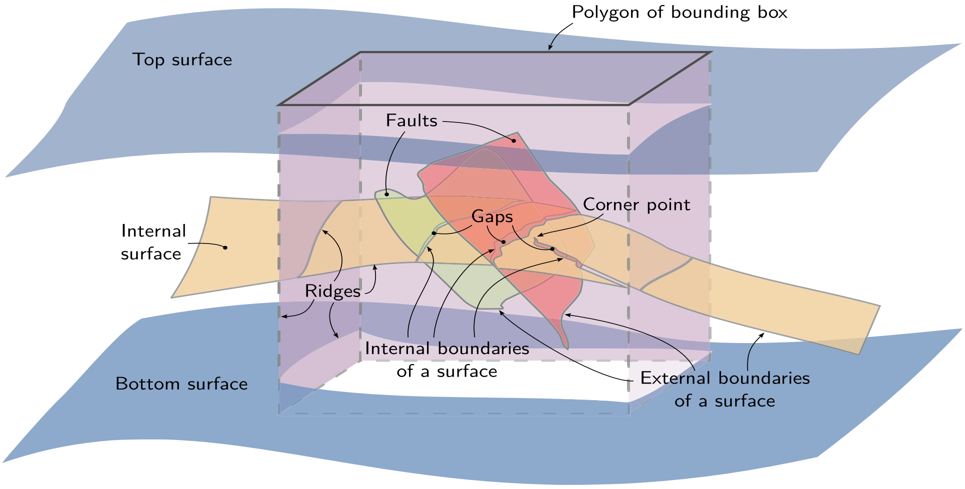


Fig. 1 Schematic of a typical domain

- Matrix containing the list of edges bounding each triangle, $\mathcal{B}^{\mathcal{D}}$.
- Linked list containing the triangles connected to each node, $\mathcal{L}^{\mathcal{D}}$, is generated.
- Edges with more than two neighbouring triangles are duplicated in the data structure and linked together using a separate linked list.

The above data structures are created for each individual surface and merged into a global data structure that contains the triangulation of the complete domain of interest.

For searching purposes alternating digital trees, ADT, are utilised [18]. An ADT storing nodes, $\mathcal{T}_p$, is built to enable the efficient identification of nodes within a given proximity. An ADT storing edges, $\mathcal{T}_e$, and an ADT storing triangles, $\mathcal{T}_t$, are utilised for efficient intersection detection. A dynamic data structures that can be modified easily and efficiently as nodes, edges, triangles are created or deleted has been adopted. Memory is allocated in a dynamic fashion that accommodates dynamic growth of the mesh. However, the link lists utilised, enable tracking of unused spaces vacated by deleted elements and prevent the need for multiple deallocation and allocation.

3.2 Open boundary and ridges

The identification of ridges and open surfaces can only be correctly carried out if the triangulation of the surfaces are consistently oriented. For each surface, the orientation is first validated by starting from any triangle and iteratively visiting the adjacent triangles to ensure the consistency of the connectivity. With consistent orientation, the edges in the data structure that are bounded by one triangle form the boundary of an open surface. These boundary edges are classified as internal and external. The external edges represent the outer boundary of the surface while the internal edges represent gaps in the surface that are normally at the location of intersecting fault surfaces. An edge can also be classified as a ridge if the dihedral angle between its two adjacent triangles is smaller than a used defined value. These types of edges will be treated as ridges and all local operations will be restricted and will only take place along the ridges. Special care has to be paid to the computation of curvature and normals at the nodes forming the open ridges. To ensure a smooth ridge representation, the curvature at each node of the ridge, aided by its two adjacent nodes on the ridge, is computed in the tangent to the surface using the Menger curvature. This results in a zero curvature for straight line boundary and increases as the angle between the two edges meeting at the node decreases. Similarly, the normal to the ridge node is calculated in the tangent plane to the surface using weighted average of the normals of the two connecting boundary edges.

At the opposite end of the spectrum are surface edges that are connected to more than two triangles. These edges will also be treated as ridges and all local meshing operations are performed along the ridges.

3.3 Surface intersection

The surface intersection assumes the presence of two triangulated surfaces $\mathcal{D}_i$ and $\mathcal{D}_j$ that are valid and not self–intersecting. It is possible to construct the intersection curve between the two surfaces $\mathcal{I}_{ij}$, by identifying intersecting pairs of triangles and subdividing them based on the type of intersection. However, this process requires a different element subdivision for each of the 8 possible types of intersection based on the location of the intersecting nodes within the triangles. Hence, the adopted process is based on the identification of edge–face intersection that requires the subdivision of the edge into two edges and the subdivision of the triangle into three triangles for an intersection node that lies inside the triangle. If the intersection node lies on the edge of a triangles, each of the two triangular elements neighbouring the edge have to be subdivided into two triangles. Adopting edge–face intersection for the construction of the intersection curve, requires the identification of the intersection of edges in $\mathcal{E}^{\mathcal{D}_i}$ with triangles in $\mathcal{C}^{\mathcal{D}_j}$ and the intersection of edges in $\mathcal{E}^{\mathcal{D}_j}$ with triangles in $\mathcal{C}^{\mathcal{D}_i}$.

Algorithm 1 describes the framework of the multiple surface intersection procedure. The provided list of surfaces that form the domain of interest are sequentially inserted. The list could include enclosed and open surfaces that represent internal interfaces, faults or domain bounding surfaces. In the current implementation, faults are inserted first. Following the validation check for orientation, the first surface triangulation in the list is treated as the base surface of the global data structure, $\mathcal{D}_g$. For each additional surface triangulation, a temporary data structure is initialised following the validation check, $\mathcal{D}_t$. Intersections between the surface triangulation present in the global data structure and the surface triangulation present in the temporary data structure is sought. If an intersection is detected, both data structures are dynamically updated to include the new and modified triangulations that results from the additional intersection nodes. It is worth noting that the edge–face intersection process account for three possible scenarios:

- the point of intersection is within the triangle: the triangles is subdivided into three triangles.
- the point of intersection is on an edge: the triangle is subdivided into two triangles and the adjacent triangles that share the edge is also divided into two triangles.
- the point of intersection coincide with a node of the triangles: the triangle does not require any subdivision.

Local mesh improvement using edge swap and edge collapse is then employed separately on each surface in the vicinity of the modified elements preserving the established intersection edge. This improvement to the mesh quality prevents errors that could occur due to very thin or small triangles if multiple surfaces intersect at the same location. Upon the completion of the intersection process, the surface triangulations present in the temporary data structure will be added to the triangulation present in the global data structure. This process is then repeated for the remaining list of surfaces. When all surfaces have been inserted, triangles that are outside the enclosed domains will be removed from the triangulation present in the global data structure. This process is driven from external open edges and iteratively marched using the connecting triangles toward intersection curves with the outer boundary surfaces.

Finally, to facilitate the identification of internal regions, it is assumed that the provided surface triangulation of a closed domain has normals that are pointing toward the domain of interest. Additionally, triangulated surfaces that are used to bound the domain also have their normals pointing toward the domain of interest. The detected intersection curves are then used to separate patches that belong to different regions. Internal surfaces are flagged to ensure that the volume mesh will be generated on both side of those surfaces.

The surface intersection is utilised for the generation of a domain bounding surface. In this case, the coordinates, in the horizontal plane, of the nodes of a user defined polygon are used to construct the bounding surfaces as shown in Fig. 1. A surface mesh on the vertical planes generated from each edge of the polygon are created and are intersected with the extracted surface meshes to form an enclosed domain suitable for volume mesh generation.

3.4 Surface extension

Different procedures are often used to extract all the geological features that are important for reservoir modelling. When these surfaces are integrated, they seldom result in a conforming surface triangulation that is free from gaps. A scenario that may occur when matching the geological inter-

Algorithm 1 Surface intersection.

```
Input: Set of $n$ surface triangulations;
Read the first surface triangulation and generate the global data structure $\mathcal{D}_g$;
Build the nodes, edges and triangles alternating digital trees, $\mathcal{T}_g^p$, $\mathcal{T}_g^e$, $\mathcal{T}_g^t$;
for $k = 2$ to number of surfaces $n$ do
    Read surface and generate temporary data structure, $\mathcal{D}_t$;
    Build the nodes, edges and triangles alternating digital trees, $\mathcal{T}_t^p$, $\mathcal{T}_t^e$, $\mathcal{T}_t^t$;
    for $i = 1$ to number of triangles in $\mathcal{C}^{\mathcal{D}_g}$ do
        Search $\mathcal{T}_t^e$ and create the list the edges, $\mathcal{S}$, that lie within the bounding box of $\mathcal{C}_i^{\mathcal{D}_g}$;
        for $j = 1$ to number of edges in $\mathcal{S}$ do
            if edge $\mathcal{S}_j$ intersect triangle $\mathcal{C}_i^{\mathcal{D}_g}$ then
                Locate intersection node;
                Break edge;
                Break triangle;
                Update data structure;
                Check for possible local edge swap on each surface;
                Check for possible local edge collapse on each surface;
            end if
        end for
    end for
    for $i = 1$ to number of triangles in $\mathcal{C}^{\mathcal{D}_t}$ do
        Search $\mathcal{T}_g^e$ and create the list of edges $\mathcal{S}$ that lie within the bounding box of $\mathcal{C}_i^{\mathcal{D}_t}$;
        for $j = 1$ to number of edges in $\mathcal{S}$ do
            if edge $\mathcal{S}_j$ intersect triangle $\mathcal{C}_i^{\mathcal{D}_t}$ then
                Locate intersection node;
                Break edge;
                Break triangle;
                Update data structure;
                Check for possible local edge swap on each surface;
                Check for possible local edge collapse on each surface;
            end if
        end for
    end for
    Merge the data structure $\mathcal{D}_t$ into the global data structure $\mathcal{D}_g$;
    Update the trees;
end for
Remove outside triangulations and form closed domains;
```

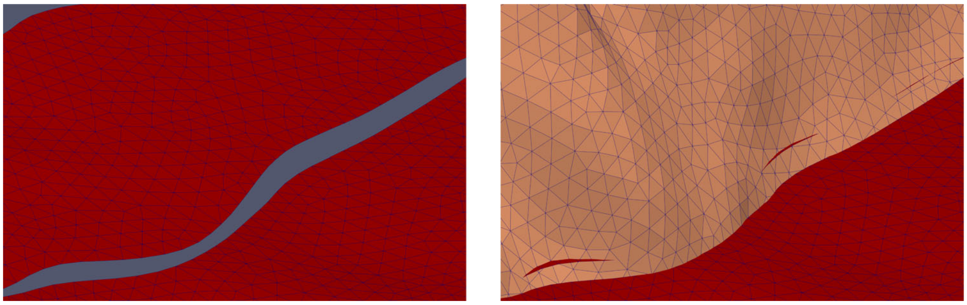

(a) Open edges of a gap in a surface (b) Discontinuous intersection with a fixed plane

Fig. 2 Internal gaps that require surface extension

face surface meshes to the faults surface meshes, is that the open edges on one geological interface surfaces do not touch or intersect the triangulation of the faults surface meshes. Figure 2 shows the interface surface triangulation that failed to intersect the faults surface mesh resulting in a broken intersection curve and internal gaps. This will result in a non conforming surface mesh and prevent the identification of the different regions within the domain. To circumvent this scenario, a surface extension procedure has been developed to ensure that no internal open edges of the geological interface surface, that lie within a prescribed user defined distance from a fault, are remaining within the domain.

The aim of this procedure is, during the intersection process, to extend a surface mesh, in the vicinity of its open edges, such that it will intersect with a surface in close proximity. The value that determines close proximity is user defined and is chosen based on the known accuracy of the used techniques in the initial extraction of the surfaces. Hence, it is possible for an open edge to be an internal edge within the domain if its nodes are further than the prescribed value of close proximity.

Here, it is assumed that the surface mesh that does not contain the open edge is fixed and present in the global triangulation $\mathcal{D}_g$ and the surface with open edges, $\mathcal{D}_t$, will be extended along the normal to these edges prior to the calculation of the intersection curve.

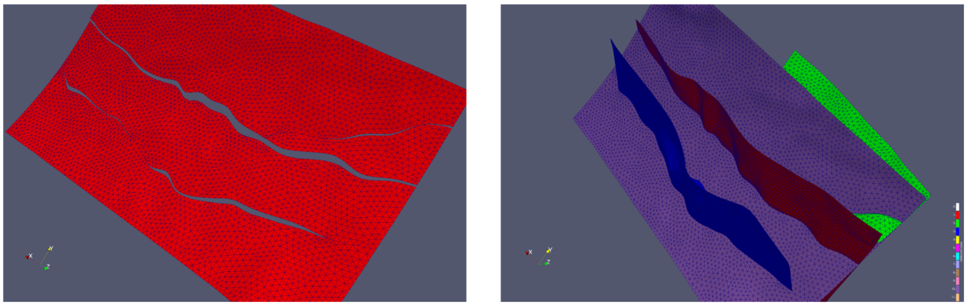
Algorithm 2 shows the general framework of the gaps closing procedure. The process starts by identifying individual loops of open edges or segments of open edges. The loops are broken into segments that connect two consecutive corner nodes. Figure 3 shows an example of a surface that contains multiple loops of open edges and segments of open edges. In a similar fashion to identifying ridges, corner nodes are determined based on the angle between the two adjacent edges.

For each segment, based on the provided value of close proximity, edges that are within the value of close proximity from the fixed surface are placed in a list, $\mathcal{L}_t$. Intersection of the surface containing the edges in $\mathcal{L}_t$ with the identified fixed surface is sought. To ensure that the intersection curves lie on the fixed surface, the surface mesh of the open edges must intersect the fixed mesh at every node forming the open edges. If no intersection is found, the surface is extended along the normal of the nodes of the open edge by the user prescribed value of close proximity. This will ensure overlap with the surface $\mathcal{D}_g$. In order to ensure the uniqueness of the identified corner nodes following the intersection process, projection onto the surface in close proximity replaces the extension. It needs to be emphasis that the process is dependent on the user prescribed value of close proximity. Open edges are allowed to be present in the final mesh. In the current implementation, the value of close proximity is prescribed per surface. This allows greater flexibility and caters for the different accuracies used to construct the geological surfaces.

Algorithm 2 Framework of surface extension program.

```
Input: The triangulation of the fixed surface, $\mathcal{D}_g$;
Input: The triangulation of the surface with open edges, $\mathcal{D}_t$;
Identify individual open segments in $\mathcal{D}_{\sqcup}$, $nSeg$;
for $i = 1$ to $nSeg$ do
    Place the list of edges on the segment $i$, that are within
    close proximity of surface $\mathcal{D}_g$ in a list, $\mathcal{L}_t$;
    for $j = 1$ to number of edges in the list do
        Check if the triangle of surface $\mathcal{D}_t$ that contains the
        edge intersect a triangle in $\mathcal{D}_g$;
        if Less than two intersection nodes between the
        triangle of surface $\mathcal{D}_t$ that contains the edge and a
        triangle in $\mathcal{D}_g$;
        then
            Extend the surface along the normal to the open
            edge;
        end if
    end for
end for
Perform Surface intersection;
```

When the surface extension is completed, the surface intersection process described in Algorithm 1 is employed to determine the intersection curve between the added surface and the current surfaces in the global mesh data structure. For all extended surfaces the portion of the mesh that lies between the computed intersection curve and the extended open edges is removed.



(a) Loops and segments of open edges

(b) Intersecting surfaces

Fig. 3 Intersecting surfaces that may have gaps

4 Surface remeshing

The aim of the surface remeshing is to improve the quality of a valid surface triangulation and to ensure that the size of the elements comply with a user defined spacing, whilst preserving the geometry. The adopted approach is based on local mesh modification operations that consist of edge splitting, edge contraction, edge swap and nodal relocation. These operations are related to normalised edge lengths with respect to the computed spacing.

4.1 Curvature controlled mesh sizing

Various techniques are available for defining the spacing within the domain of interest [19]. Two of the most effective techniques that are based on the use of a background mesh or collection of sources [20] have been utilised. In addition, since most numerical simulations require higher resolution in the vicinity of rapid geometric changes, a curvature-controlled metric has also been used for computation of the spacing to ensure that geometric accuracy is achieved. Sizing derivation from curvature has been well developed in several papers [21–24]. The main radii of the curvature are used to construct a geometric metric that is used to prescribe mesh sizes. For isotropic meshing, the largest radius is used. For the computation of the curvatures at a surface node, a local quadric surface is constructed based on a least-squares fit using the neighbouring mesh nodes [21]. The curvature can then easily be derived from the computation of the fundamental forms of the this quadric. Using the computed spacing, the edge normalised length is computed [11]. A normalised edge length, with respect of the spacing required at the location of the edge mid–point, between $C_s = \sqrt{2}$ and $C_c = 1/\sqrt{2}$ is considered acceptable. Edges having a normalised length larger than C_s are split, and edges with a normalised length smaller than C_c are collapsed. In many scenarios, the spacing required to resolve the physics could be much larger than the geometric constraint. Using such spacing will result in degenerate elements that affect the quality of the solution. The use of sources enables the user to locally control the spacing of the required meshes, producing good quality elements and preventing the occurrence of an ill-condition system at the solution stage.

For any internal surface node, i.e. non ridge node or corner node, its normal is computed using the area weighted average of the normals of its connecting triangles. For a ridge node or a corner node, multi-normals are used. The triangles connecting the node are divided into several patches separated by the ridges. Using the triangles forming each patch, a weighted average normal is computed. These multiple treatments of normals and curvatures ensure that the refinement process will produce meshes which conform accurately to the given geometry.

To avoid a sudden jump in the size of the elements when moving from regions of high curvature to regions with low curvature, gradation of the curvature based spacing is carried out. A user defined gradation parameter per unit length is employed to control the rate of change of the size of the elements. An iterative process is employed to ensure that the spacing at adjacent nodes does not grow at a rate larger than the user prescribed gradation factor. In the current implementation a gradation factor of 15% with 10 iterations produces smooth meshes.

4.2 Edge splitting

Edges that have a normalised edge length larger than the given splitting criterion C_s, are divided. A new node is located on the surface using the G1 interpolation [20, 25] to split the edge into two smaller edges. The process starts by placing the node at the mid–point of the edge. The node is then projected onto the surface. An iterative process is utilised to relocate this node in order to ensure that the subdivided edges have equal length in the physical space. The two adjacent triangles to the edge under consideration will be replaced by four new triangles and all the data structures will be updated accordingly. Following the node insertion, local edge swapping that only involves the newly created triangles is performed to improve the geometry approximation, the element shape quality and the robustness of the procedure.

For ridge edges, the multi-normals are used to project the node onto the adjacent surfaces using the G1 interpolation resulting in multiple physical set of coordinates. The final physical coordinate is computed using a weighted average based on the curvature of the adjacent surfaces. The weighting ensures that the nodes will be moved toward the surface with high curvature.

4.3 Edge collapse

If the normalised edge length is less than the collapse criterion C_c, the edge will be removed by merging the two nodes forming the edge under consideration. The merged node can be either one of the two end nodes or a new node located along the edge to be removed. The quality of elements that results from collapsing the edge toward one of its end nodes or its mid node is evaluated and the configuration with the best quality is adopted.

If any edge has one of its nodes on a ridge, collapse will be carried out toward the ridge node. Special care is needed when collapsing an edge that has both its end nodes on a ridge. In this case, the collapse can only take place if the two nodes are consecutive nodes on the same ridge.

In addition to the validity check to ensure that no self intersecting triangles will result from the edge collapse procedure, local edge swapping is employed for elements connected to the displaced node. For an efficient validity check, the triangles ADT data structure, $\mathcal{T}_t$, is utilised to identify the triangles in the vicinity of the edge under consideration that could be affected by the collapse operation.

4.4 Edge swapping

The objective of edge swapping is to maximise a chosen quality measure of the pair of triangles that share the edge. Different measures can be adopted to analyse the quality of a configuration. These can be classified as being either measures of element shape or measures of geometric approximation.

The first geometric approximation measure [21, 26] is the gap between the surface and the triangle. This is defined as the maximum angle between the triangle normal and the normals evaluated at the nodes of the triangles. The second measure is the smoothness criterion. This is defined as the angle between the normal of two adjacent triangles. Various element quality measurements can be used for the element shape analysis such as minimum angles and number of triangles surrounding a node [19, 27].

In the swapping procedure, the validity of the surface triangulation must be maintained. This is accomplished by using procedures similar to those employed when edge collapse is performed.

In the current implementation, in the initial refinement and coarsening stage, the geometry approximation quality indicator is used to maintain the geometric features. A final optimisation loop that uses element quality is performed to ensure smoothness of the final mesh.

4.5 Nodal smoothing

Node smoothing, or relocation, modifies the position of a node without changing the topology of the triangles meeting at the node. The objective is to find an optimal position on the surface for the node, in the sense that this alternative configuration should have better quality, according to criteria such as minimal angle, shape quality, size conformity or others [19]. Here, only the element shape quality is considered and various smoothing methods can be used [19, 22, 28].

A Gauss-Seidel iterative technique is utilised for the nodal smoothing precess. The smoothing takes place in the tangent plane constructed at the node under consideration [26]. Using the spring analogy method, a new node is located at the centroid of all the projected neighbouring triangles. The triangle which contains the new node is identified and the node is projected onto the surface using G1 interpolation. Finally, the triangulation which will result from replacing the initial node by the new node is evaluated using the same geometric and shape measures used in the edge swapping. In order to ensure a robust implementation of the smoothing procedure, a relaxation technique is implemented. A node located on a ridge is only smoothed using the two adjacent nodes on the same ridge. Ten Gaus-Seidel iterations have been used for all the examples shown in the paper.

4.6 Refinement/coarsening procedure

The refinement/coarsening procedure iteratively modifies an existing surface triangulation through edge splitting or contraction. The objective is to ensure that elements are in better conformity with the size specification. Edge lengths computed with respect to size or metric are compared with the given splitting parameter C_s and contraction parameter C_c.

If the initial mesh is almost isotropic, sequential processing of the edges of the current mesh is feasible and efficient. However, the initial surface triangulation may inherit stretched elements from the anisotropic structured grid used to extract the geometry from seismic data. The sequential use of the list of edges in the split and contraction operation, may result in highly distorted local configurations that prevent further operations. To overcome this problem, larger edges are considered first. This has been shown to produce a more stable and robust procedure for refinement/coarsening of the initial mesh.

When the normalised edge length for all edges fall within the allowable limit, a final optimisation loop that consists of a combination of edge swapping and node smoothing is carried out. At this stage, edge swapping will be based on the valence measures which represents the number of edges connected to a node. A nodal value of 6 is the ideal valence for an isotropic mesh on a flat surface. For a ridge, the valence at each node is computed separately for the two adjacent surfaces based on the angle between the two edges on the ridge at the node. This procedure has proved to produce smoother meshes that are better suited for numerical modelling. Algorithm 3 describes the framework of the surface remeshing procedure.

It is worth noting that, by default, surface remeshing is carried out following a surface intersection procedure. This will ensure good quality surface meshes in the vicinity of the intersection and will prevent the generation of degenerate triangular elements if multiple surfaces intersect in close proximity. In addition, the surface remeshing could be activated prior to surface intersection if the resolution of the two surfaces differ by more than an order of magnitude. This will ensure a good quality triangulation that better conforms to the underlying surfaces.

Algorithm 3 Framework of surface remeshing program.

```
Input: The triangulation of the surface, $\mathcal{D}_g$;
Create the data structure $\mathcal{C}^{\mathcal{D}}$, $\mathcal{E}^{\mathcal{D}}$, $\mathcal{B}^{\mathcal{D}}$,$\mathcal{L}^{\mathcal{D}}$, $\mathcal{T}_p$,$\mathcal{T}_e$,$\mathcal{T}_t$;
Check mesh consistency;
Calculate mesh metrics;
Calculate normalised edge length, $\mathcal{L}_i$;
Compute the number of iteration required
$N_{iter} = int(max(max\{\mathcal{L}_i/\sqrt{2}\}, min\{\mathcal{L}_i * \sqrt{2}\})) + 1$;
for $iter = 1$ to $N_{iter}$ do
    for $j = 1$ to number of edges do
        if $\mathcal{L}_j < 1.0/\sqrt{2}$ then
            Collapse edgee;
            Update data structure;
            Local edge swapping for elements connected to
            the nodes of the edge under consideration;
        end if
        if $\mathcal{L}_j > 1.0/\sqrt{2}$ then
            Split Edge; Update data structure;
            Local edge swapping for elements connected to
            the nodes of the edge under consideration;
        end if
    end for
end for
for $iter = 1$ to $N_{Smooth}$ do
    Perform nodel smoothing;
    Calculate normalised edge length, $\mathcal{L}_i$;
    for $j = 1$ to number of edges do
        if $\mathcal{L}_j < 1.0/\sqrt{2}$ then
            Collapse edgee;
            Update data structure;
        end if
        if $\mathcal{L}_j > \sqrt{2}$ then
            Split edge;
            Update data structure;
        end if
    end for
    Global edge Swanpping;
end for
```

5 Volume mesh generation

The volume formed by the closed triangular surface mesh produced using the described procedures, will be filled with tetrahedral elements using the in house FLITE meshing suite. The technique is based on the Delaunay triangulation with automatic node creation[29]. The procedure starts by the creation of a tetrahedral mesh from the boundary nodes that ensures the existence of all boundary triangles. Various techniques can be used to control the size of the elements within the domain. Background meshes and sources can be utilised to control the spacing, however, in their absence, linear interpolation of the spacing at the boundaries is used to ensure a smooth variation of the size of the elements within the domain. Mesh enhancement techniques are also employed to improve the quality of the final mesh. This includes multiple loops of edge swapping and nodal relocation. A consistent mesh on both side of any internal surfaces is ensured. The final stage of the process is the grouping of the elements into individual domains. This process starts from an external boundary surface and iteratively groups elements that do not cross a different boundary surface.

6 Examples

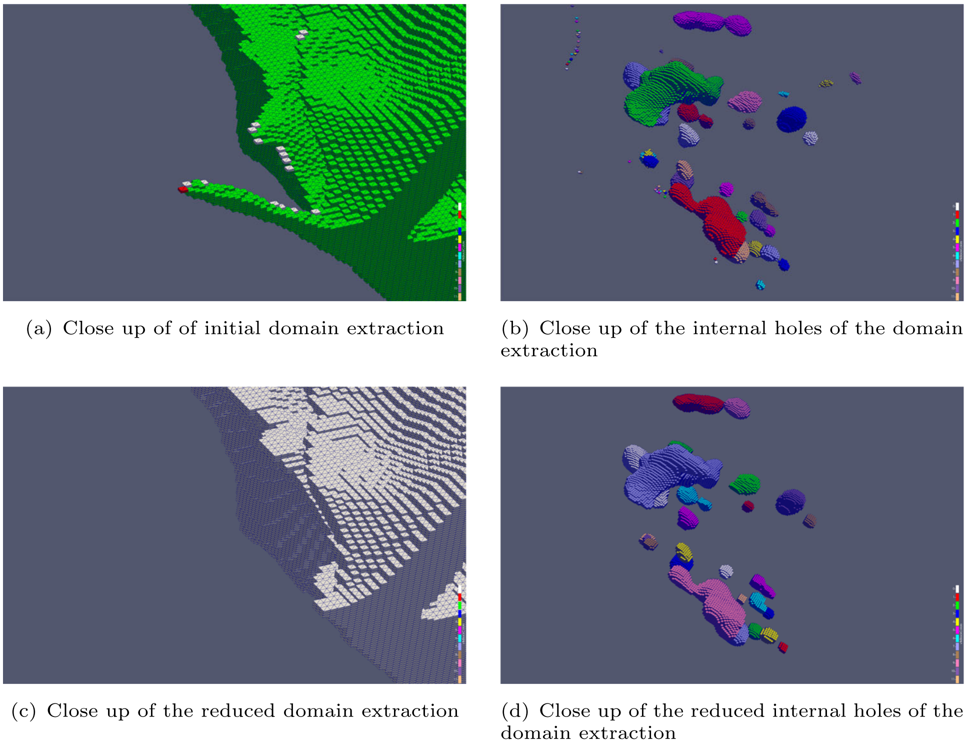
This section gives the details of 3 examples that are used to demonstrate the applicability of the procedures developed on industrially related geological data. In addition, one numerical simulation example of a industrial case study is shown to demonstrate the advantages that the system offers to control the elements' quality on the solution accuracy.

6.1 Unstructured domain extraction from seismic data

The first example deals with the generation of an unstructured volume mesh starting from high resolution seismic data. The available seismic data contains $500 \times 500 \times 1000$ nodes. The nodes are spaced at 100m in the horizontal plane and at 25m in the vertical plane. At the extraction stage it was assumed that features that are less than 200m wide in the horizontal plane or less than 50m thick in the vertical plane are insignificant and can be removed. This was applied to externally disconnected features and internal holes. Figure 4(a-b) shows a close up of some local features that were classed as insignificant prior to their removal. Figure 4(c-d) shows the same location after these features were eliminated. The extraction process resulted in two large disconnected domains. The dip and the azimuth were used to orient the outer surface of the domain as can be seen in Fig. 5.

The surface mesh consisted of 1.2 million triangles and 600K nodes. The remeshing process was then triggered to generate a modified surface mesh that has a spacing of 100m in regions of high curvatures and 500m on flat regions. The technique results in a surface mesh that consisted of 1.19 million triangular elements and 595K boundary nodes. Figure 6(a) shows the surface mesh obtained following the remeshing procedure. The surface mesh consisted of 252K elements and 126K nodes. The distribution of the value of the normalised edge length is shown in Fig. 7(a). Geometric restriction has resulted in some edges having their normalised edge length outside the expected limit of $[1.0/\sqrt{2}, \sqrt{2}]$. The distribution of the ratio of minimum edge length to maximum

Fig. 4 Domain extraction from seismic data



(a) Close up of of initial domain extraction

(b) Close up of the internal holes of the domain extraction

(c) Close up of the reduced domain extraction

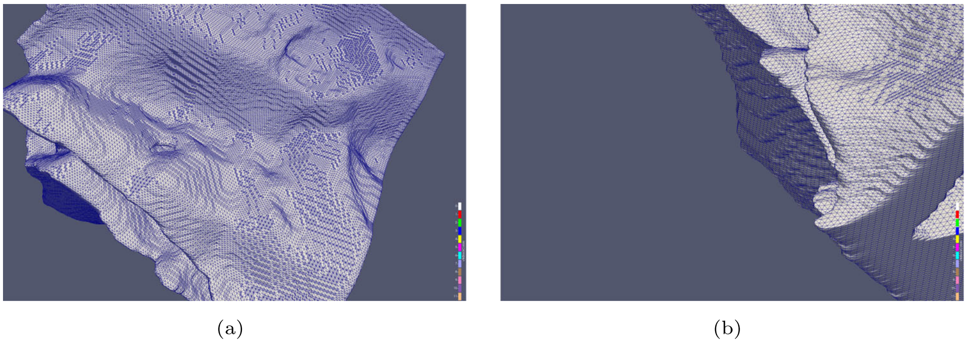
(d) Close up of the reduced internal holes of the domain extraction

edge length at a point is shown in Fig. 7(b). It is expected that due to the use of curvature control for mesh spacing, and the geometric restriction, that prevents elements from crossing identified ridges, the ratio of maximum to minimum edges at those points would be smaller than the theoretical limit of [0.5 , 1.0]. The distribution of the minimum angle in a triangle is shown in Fig. 7(c). It is also expected that mesh gradation, which is required to smooth the local spacing at high curvature regions, induced the minimum angle of some triangles to fall outside the theoretical limit, that is based on the range of acceptable edge lengths, of [30° , 60°].

A tetrahedral mesh was finally created starting from the remeshed extracted surface. The volume mesh consists of 33.8 million tetrahedra and 5.56 million nodes. A cut through the mesh can be seen in Fig. 6(b). The minimum dihedral angle of the tetrahedra was used as an indicator to the quality of the generated volume mesh. As can be seen in Fig. 8(a), very good mesh quality can be observed. The minimum dihedral angle was 11.5° which resulted from geometric restriction. A very small percentage of elements had dihedral angle below 30.0°. The normalised edge length was used to indicate the compliant of the generated elements with the user defined mesh spacing. Figure 8(b), shows the distribution of the normalised edge length over the domain. Over 99% of the edges have a normalised edge length within the permitted region of $[1.0/\sqrt{2}\,,\,\sqrt{2}]$. The exemption was found to be due to geometric restriction that prevented the edge collapse. It should also be noted that following final nodal smoothing it is possible for some edges the slightly exceed the allowable limit.

To demonstrate the robustness of the technique to include additional features that were separately extracted, three planes that intersect the domain at different locations have been added to the extracted surface. Two of the selected

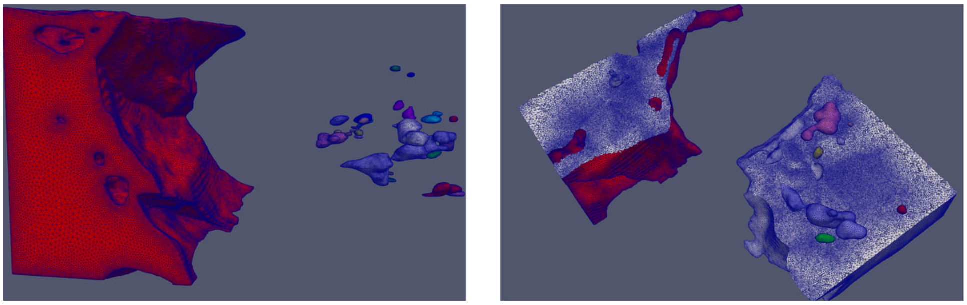
Fig. 5 Close up of the extracted surface



(a)

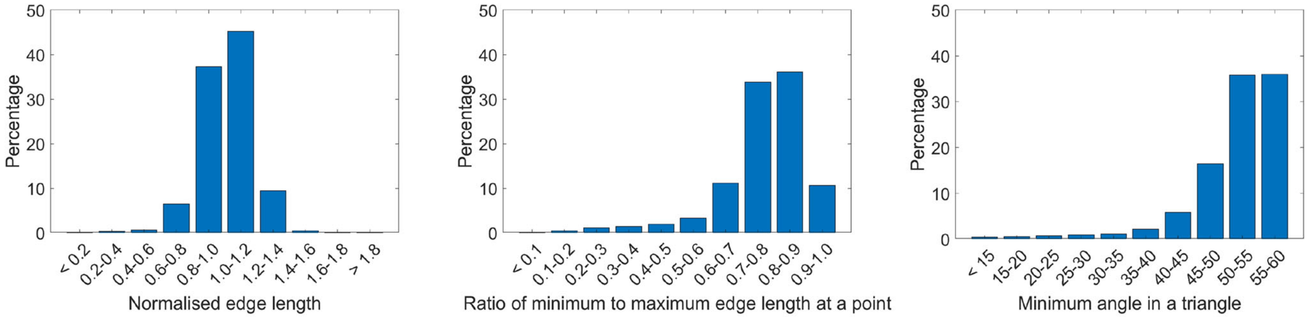
(b)

Fig. 6 Surface and volume mesh



(a) Close up of the curvature controled remeshed surface

(b) Close up of the curvature controled volume mesh



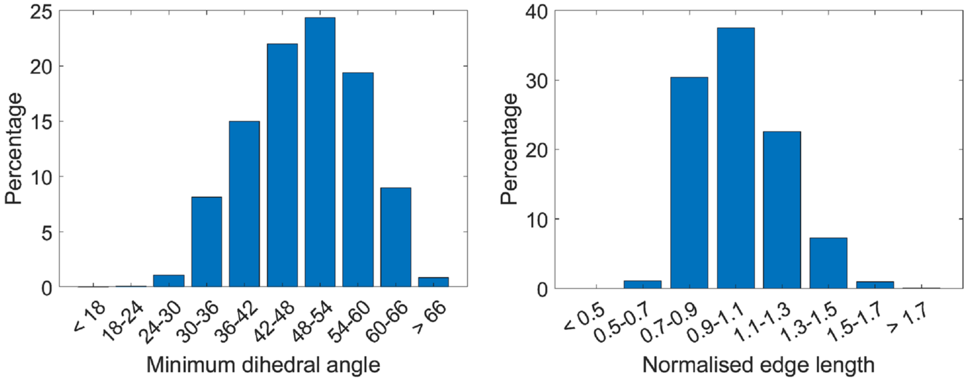
(a)

(b)

(c)

Fig. 7 Analysis of the quality of the extracted surface mesh

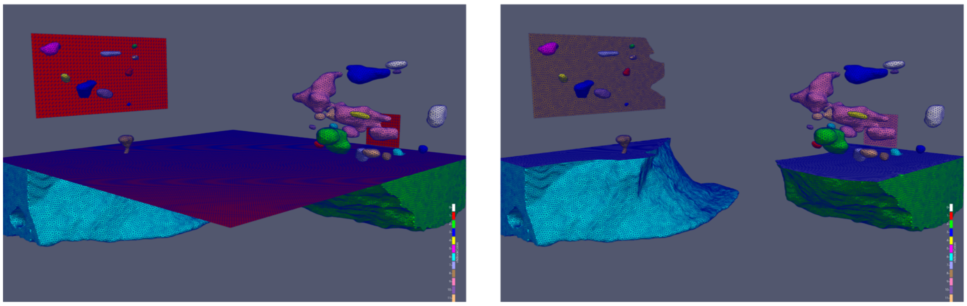
Fig. 8 Analysis of the quality of the volume mesh



(a)

(b)

Fig. 9 Insertion of 3 user defined planes



(a) Surface mesh of the added planes

(b) Final surface mesh of intersecting planes

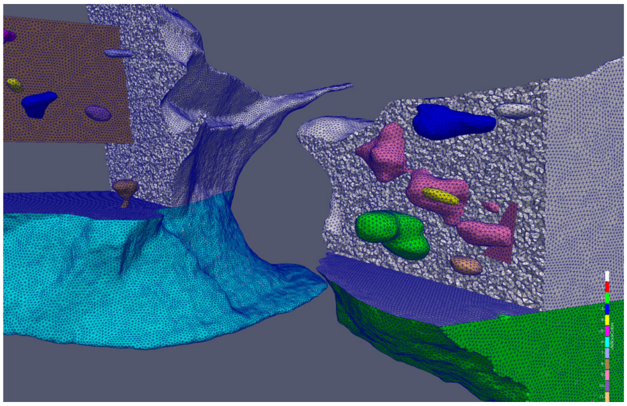


Fig. 10 Volume mesh with the inserted planes

planes intersect with local features of the extracted domain while the third plane cut across the complete domain. A uniform spacing of 250m was specified for the final mesh. Figure 9(a) shows the location and size of the 3 planes in relation to the extracted domain prior to the intersection and remeshing process. Figure 9(b) shows the final surfaces after the intersection and the remeshing process.

The generated volume mesh that preserves the internal surfaces is shown in Fig. 10. The volume mesh consisted of 8.4 million tetrahedral elements and 1.45 million nodes.

Finally, to demonstrate the effect of grid reduction during the extraction stage. The mesh was re-generated by merging groups of three cuboids in x and y directions and groups of four in the z direction. This is equivalent to having a data resolution of 300m in the horizontal plane and 100m in the vertical plane. Isolated features that have any dimensions below these values are assumed insignificant in this case and can be removed. During the surface remeshing, regions of high curvatures were assigned spacing of 200m while regions with low curvatures were assigned spacing of 600m. The surface mesh consisted of 73.4K triangular elements and 36.6K nodes.

The volume mesh consisted of 2.3 million tetrahedral elements and 396K nodes. Figure 11 shows the remeshed surface and a cut through the volume.

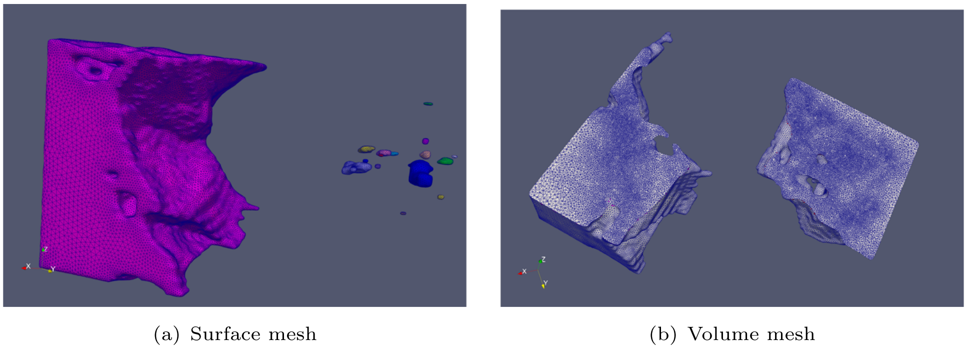
It is clear that, due to the reduction in resolution of the seismic data, a few small features have been lost and that changes to the surface details have occurred. However, the reduction of the number of elements that was achieved will result in a faster modelling time. This could be beneficial at the initial stage of exploration. It is worth noting, that smaller meshes can also be generated by increasing the size of the elements that are generated from the geometry extracted using the full seismic data. However, it was observed, as expected, that the volume meshes obtained exhibited lower quality in locations where the extracted detailed features imposed constraints on the volume mesh.

6.2 Unstructured volume mesh from multiple individual surfaces

The second example demonstrates the use of the developed techniques for the creation of an unstructured volume mesh starting from multiple geological layers that were separately extracted and triangulated. The provided data consisted of six separately provided surface meshes that represent 5 material interfaces and one fault that intersect some of the interface surfaces. In addition, a bounding box was specified to truncate the domain to encompass the region of interest. Figure 12(a) shows the surfaces with the limits of the bounding box.

The surface intersection procedure was utilised to determine the intersection of any two surfaces. This was followed by the surface remeshing to ensure that the size of all elements comply with the uniform user defined spacing of 100m. The surface mesh of the truncated domain consisted of 75K triangular elements and 37K nodes. Figure 12(b) shows a cut through the final surface mesh of the truncated domain. The distribution of the value of the normalised edge length is shown in Fig. 13(a). Over 99% of the normalised edge length fall within the permitted limit of $[1.0/\sqrt{2}\,,\sqrt{2}]$. The distribution of the ration of minimum edge length to maximum edge length at a point is shown in Fig. 13(b). Again over 99% of the values fall within the expected values of [0.5 , 1.0]. The distribution of the minimum angle in a triangle is shown in Fig. 13(c). It can also be seen that over 99% of the elements

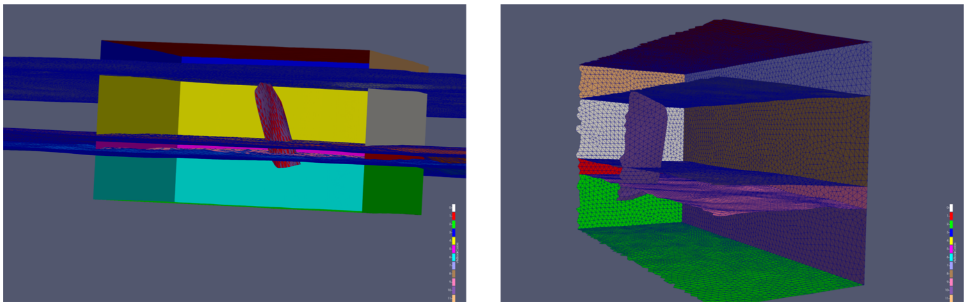
Fig. 11 Generated mesh from reduced seismic data



(a) Surface mesh

(b) Volume mesh

Fig. 12 Multi–layered domain



(a) Provided interface surfaces

(b) Provided surfaces with limits of bounding box

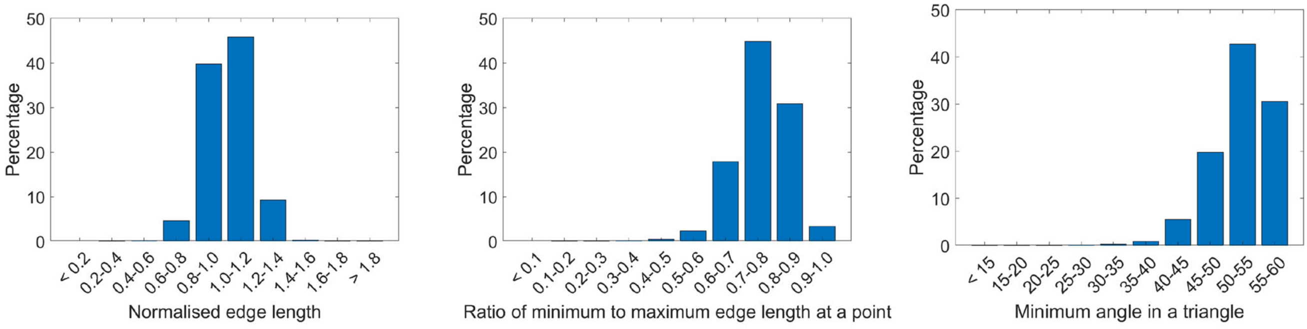
exhibit excellent quality with a minimum angle that is within the anticipated region of [30° , 60°].

The enclosed domain was filled with 276K nodes forming 1.63 million tetrahedral elements. Figure 14 shows a cut through the volume mesh. The total time taken was 6 minutes and 15 secs

6.3 Automatic repair of gaps between multiple surfaces

The final example demonstrates the ability of the developed technique to detect and close internal gaps that prevent watertight domains being constructed. Eight surfaces that represent material interfaces and three surfaces representing faults have been provided. In this example, the interface layers contain gaps that are due to the change in the level of the interface that resulted from fault formation. The surface of the three provided faults are in close proximity to the gaps but not always touching or intersecting the interface layers. Figure 15(a) shows one of the provided interfaces and fault surfaces. Similar to the previous examples, the first surface will be taken as the base surface and the remaining surfaces will be introduced one at the time. However, surfaces that contains internal open edges and gaps, i.e interface layers, should be introduced after the introduction of the fixed surfaces, i.e faults. For each introduced surface, open segments are identified and intersection with the previously inserted surface will be sought using the proposed procedure. In places where the open segments fail to intersect with a surface in close proximity, surface extension is carried out for the intersection to take place. Figure 15(b) shows a region where the open edges of the upper part of an interface layer do not cross the fault for the intersection to take place.

Figure 16(a) shows that the complete intersected surfaces following the insertion of all surfaces. It is clear that the mesh quality in the vicinity of the intersection curves contains degenerate triangles with small angles and large ratio of edge length. This quality will drastically impinge of the accuracy and efficiency of the thought simulation. Hence, the surface remeshing technique was employed to produce the final surface mesh shown in Fig. 16(b). The final surface mesh consisted of 92K triangular elements and 45K nodes. The distribution of the value of the normalised edge length is shown in Fig. 17(a), together with the distribution of the ration of minimum edge length to maximum edge length at a point, Fig. 17(b), and the distribution of the minimum angle in a triangle, Fig. 17(c). It can also be seen that over 99% of the elements exhibit excellent quality and deterioration of the quality measure was always due to geometric restriction. The total time taken was 2 minutes and 30 secs



(a)

(b)

(c)

Fig. 13 Analysis of the Multi–layered surface mesh quality

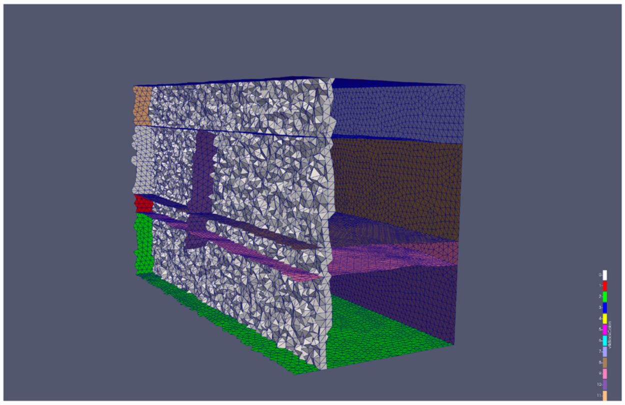


Fig. 14 Volume mesh of the multi surface domain

6.4 Impact of meshing quality on numerical simulation results

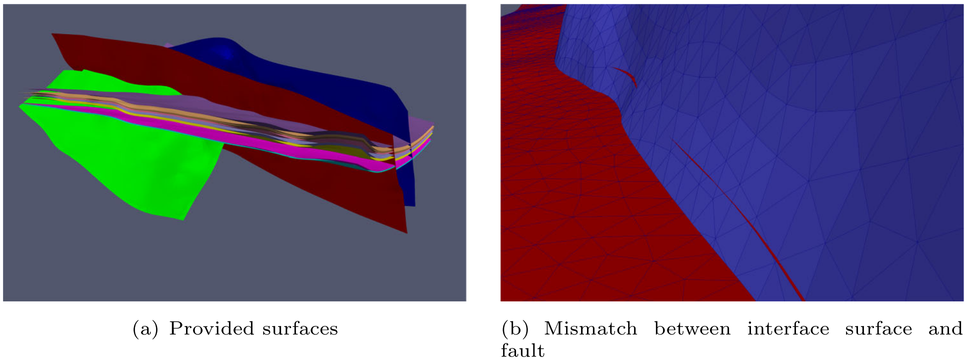
It is well studied that finite element analysis based numerical simulations will exhibit a certain degree of meshing dependency. To illustrate the importance of meshing to numerical simulation and to further validate the suitability of the generated mesh to be used for modelling complex domains, a 3D model that utilises three of the horizontal surfaces shown in Section 6.3 and the three faults from Section 6.3 with the addition of an overburden, a base surface and four surfaces to bound the domain, Fig. 18(a), is created. The resultant 3D volume mesh consisted of five regions, shown in Fig. 18(b), representing $Overburden$, $SandA$, $Shale$, $SandB$ and $Underburden$ domains. The average thickness of the overburden layer and underburden layer is 1000m. The material properties of the five domains are shown in Table 1. The normal stiffness, shear stiffness and the friction coefficient for the faults are taken to be 10GPa, 8GPa and 0.6 respectively.

The finite element solver for coupled simulation of the evolution of geological structures, $ParaGeo$, was utilised for modelling gravitational loading. The initial stress condition for the model is setup as uniaxial boundary conditions with zero displacement at all the sides. The initial pore pressure is assumed to be hydrostatic for all the layers, as shown in Fig. 19(a). To simulate the depletion induced fault slip, the $SandB$ layer is assumed to be the production layer and the pore pressure of this entire layer has dropped from average 70MPa to 45MPa after 10 years production, Fig. 19(b). The compaction of the reservoir layer will result in slip on the fault surfaces.

To illustrate the importance of mesh control to geotechnical study, two meshes have been generated. The first mesh had spacing that varied between 20m and 100m with a rapid transition from the small mesh spacing used to discretise the intersection regions of the faults to the spacing used within the domain. A finer second mesh that had a spacing variation between 20m and 40m with slow mesh gradation near the faults was generated. Figure 20 shows the meshes on fault 1 and fault 3. It can be observed that inconsistent refinement of the individual surfaces have resulted in a large variation in the aspect ratio of adjacent elements in the vicinity of the intersection with the horizontal surfaces. Figure 21(a-c) shows the pore pressure distribution on the three faults after depletion where the low pore pressure areas are embedded in the production $SandB$ layer.

The depletion induced slip on the faults computed on both meshes are compared in Fig. 22. It can be observed that despite the validity of the first mesh, the fast gradation in the size of the elements has resulted in a local concentration of high slip away from the production region. Reducing the mesh spacing and the gradation in the vicinity of the faults and the horizontal surfaces, have resulted in slips located around the production region that are depended on the reservoir depletion and fault geometry.

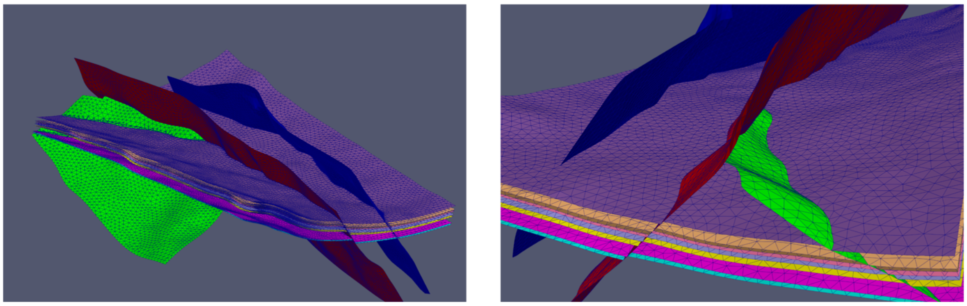
Fig. 15 Multi–layered surface with gaps



(a) Provided surfaces

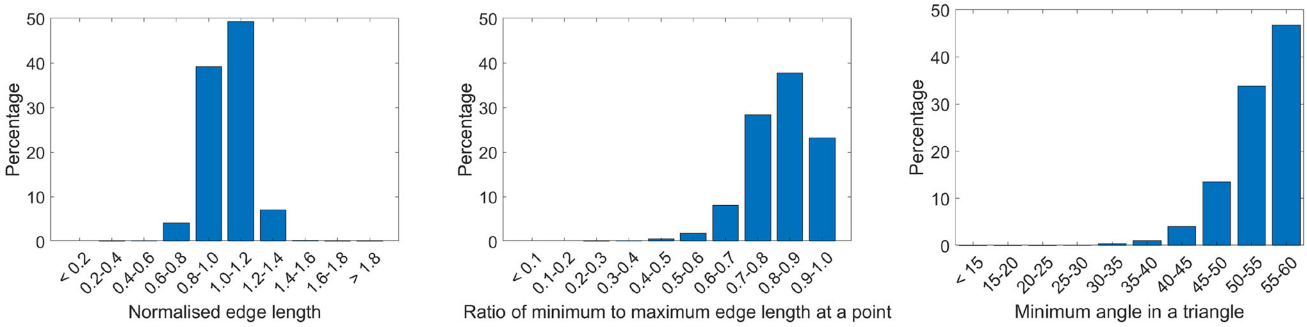
(b) Mismatch between interface surface and fault

Fig. 16 Multi–layered surface with gaps



(a) Surface prior to remeshing

(b) Surface after remeshing



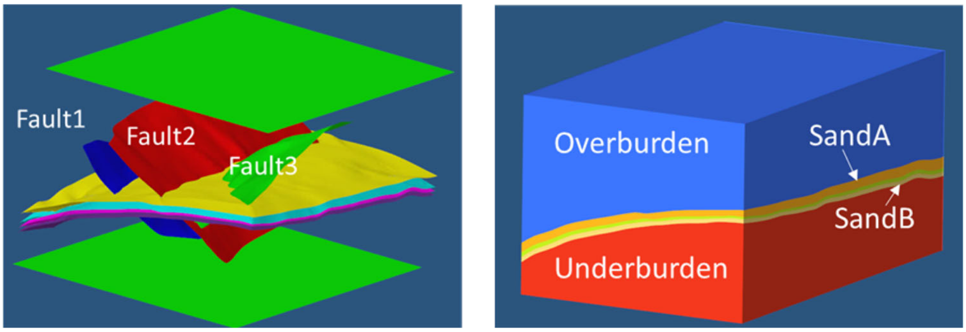
(a)

(b)

(c)

Fig. 17 Analysis of the Multi–layered surface mesh quality

Fig. 18 Separating surfaces and resulting multi-domains

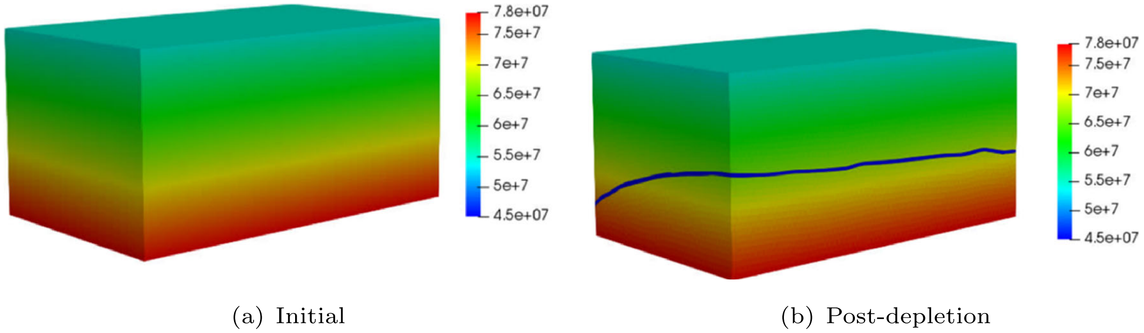


(a) 2D surfaces and faults

(b) 3D multi-domains

Table 1 Material properties

Domain	Density(kg/m^3)	Porosity	Elastic Modulas (GPa)	Poison Ration
$Overburden$	2970	0.3	64	0.18
$SandA$	2650	0.15	10	0.25
$Shale$	2650	0.02	8	0.35
$SandB$	2650	0.15	10	0.25
$Underburden$	2970	0.3	64	0.18

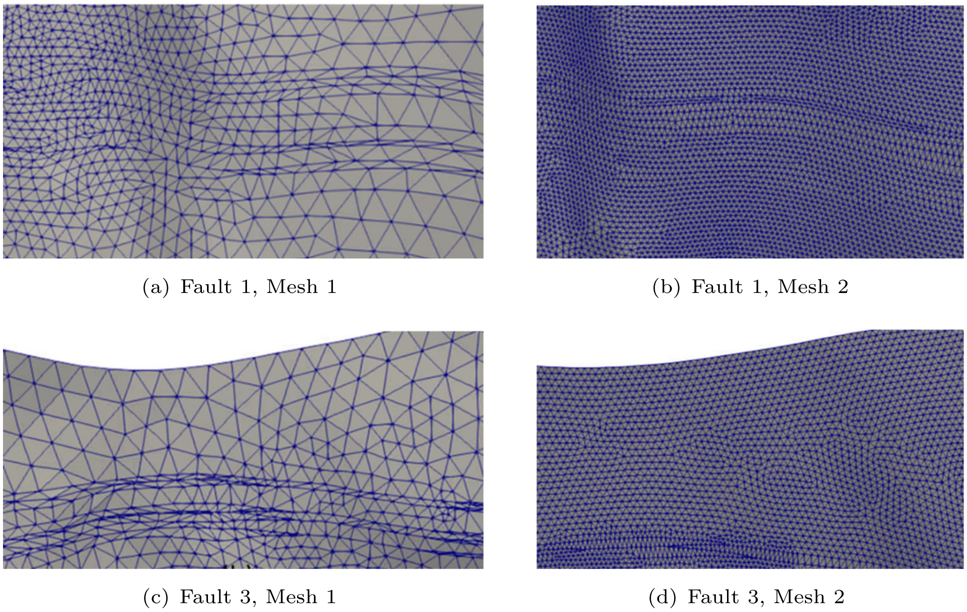


(a) Initial

(b) Post-depletion

Fig. 19 Pore pressure

Fig. 20 Close up of the surface triangulation of Fault 1 and Fault 3

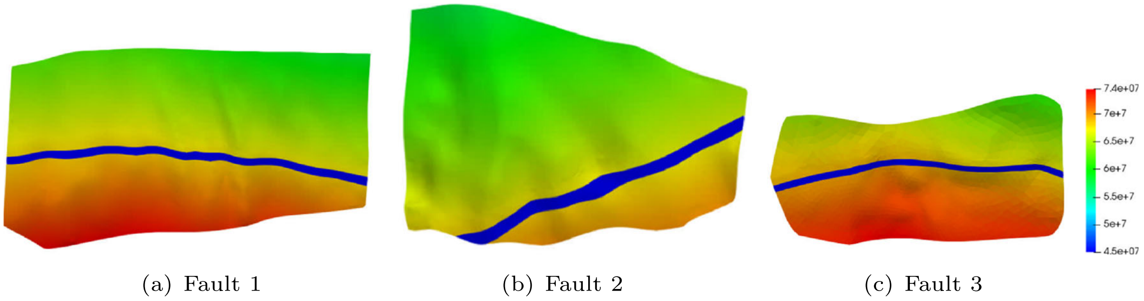


(a) Fault 1, Mesh 1

(b) Fault 1, Mesh 2

(c) Fault 3, Mesh 1

(d) Fault 3, Mesh 2

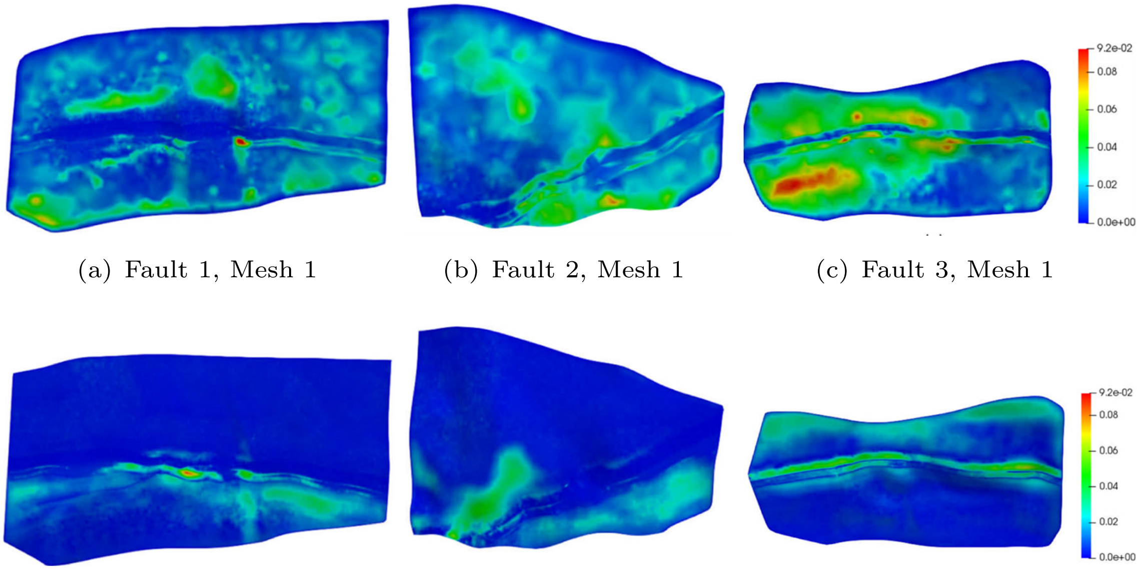


(a) Fault 1

(b) Fault 2

(c) Fault 3

Fig. 21 Post depletion Pore Pressure on the faults



(a) Fault 1, Mesh 1 (b) Fault 2, Mesh 1 (c) Fault 3, Mesh 1

(d) Fault 1, Mesh 2 (e) Fault 2,Mesh 2 (f) Fault 3,Mesh 2

Fig. 22 Computed depletion induced slip

7 Conclusion

This work presents a novel technique tailored for reservoir modelling. The process is able to start from seismic data to extract the desired surfaces, a set of extracted surfaces representing horizon and faults or a combination of both representation. This starting point will cater for the majority of the meshing requirements that is employed in industry. The implemented process is fully automated and only requires the user specification of the control parameters.

The developed technique includes the ability to close geometric gaps that occur from the different extraction techniques that are available to construct the interfaces and fault. The process requires a user specified tolerance below which two surfaces are expected to intersect. The surface extension procedure is then utilised to close the gap and ensure intersection occurs between the two surfaces.

Surface remeshing is incorporated to enhance the quality of the non-manifold surfaces. This process is user controlled and can be utilised pre and post the individual surface incorporation into the domain of the interest. The procedure produces triangulations that comply with the spacing control techniques defined by the user that include, curvature control sizing, background mesh and sources.

A facility to generate domain bounding surfaces was added to allow the creation of an enclosed domain. A Delaunay based mesh generator is then utilised to fill the domain with tetrahedral elements that also comply with the user defined spacing functions and preserving the non-manifold surface triangulation

A set of examples, that are based on industrially relevant geometries, has been presented to demonstrate the potential and the robustness of the developed technique. A final numerical example that uses one of the meshed test cases was also included to demonstrate the validity of the meshes and the importance of the available mesh control techniques to generate meshes that are capable of capturing the physics.

Funding This study was funded by Chevron Corporation.

Availability of data and material Meshes used in 6.1 and 6.2 are available at https://doi.org/10.5281/zenodo.8364469 .
Each zip file contains the input meshes and resultant surface mesh and volume mesh.
All mesh files are in .inp format.
Meshes used in 6.3 is not publicly available due to industry reasons.

Declarations

Conflicts of interest Sui Bun Lo has received research support from Chevron Corporation.
Oubay Hassan and Jason Jones have no relevant financial or non-financial interests to disclose.
Xiaolong Liu, Nevan C Himmelberg and Dean Thornton are employees of Chevron Corporation.

References

1. Zain-Ul-Abedin, M., Henk, A.: Building 1d and 3d mechanical earth models for underground gas storage-a case study from the molasse basin, southern germany. Energies. **13**(21), (2020)
2. Bentley, M.: Modelling for comfort? Petroleum Geoscience. **22**(1), 3–10 (2015)
3. Waggoner, J.R., Cominelli, A., Seymour, R.H., Stradiotti, A.: Improved reservoir modelling with time-lapse seismic data in a gulf of mexico gas condensate reservoir. Pet. Geosci. **9**(1), 61–71 (2003)
4. Yin, Z., Feng, T., MacBeth, C.: Fast assimilation of frequently acquired 4d seismic data for reservoir history matching. Comput. Geosci. **128**, 30–40 (2019)
5. Singh, V., Srivastava, A.K., Tiwary, D.N., Painuly, P.K., Chandra, M.: Neural networks and their applications in lithostratigraphic interpretation of seismic data for reservoir characterization. Lead. Edge **26**(10), 1244–1260 (2007)
6. Caumon, G., Lepage, F., Sword, C.H., Mallet, J.-L.: Building and editing a sealed geological model. Math. Geol. **36**, 405–424 (2004)
7. Jackson, M.D.D., Percival, J.R.R., Mostaghimi, P., Tollit, B.S.S., Pavlidis, D., Pain, C.C.C., Gomes, J.L.M.A.L.M.A., El-Sheikh, A.H.H., Salinas, P., Muggeridge, A.H.H., Blunt, M.J.J.: Reservoir modeling for flow simulation by use of surfaces, adaptive unstructured meshes, and an overlapping-control-volume finite-element method. SPE Reserv. Evaluation Eng. **18**(02), 115–132 (2015)
8. Zhang, Z., Yin, Z., Yan, X.: A workflow for building surface-based reservoir models using nurbs curves, coons patches, unstructured tetrahedral meshes and open-source libraries. Comput. Geosci. **121**, 12–22 (2018)
9. Seo, D., Jung, H., Sung, W.-K., Nam, D.: Development of the korean spine database and automatic surface mesh intersection algorithm for constructing e-spine simulator. J. Appl. Math. (2014)
10. Khan, D., Yan, D.-M., Gui, S., Lu, B., Zhang, X.: Molecular surface remeshing with local region refinement. Int. J. Mol. Sci. **19**(5)
11. Frey, P.J.: About surface remeshing (2000)
12. Löhner, R.: Regridding surface triangulations. J. Comput. Phys. **126**(1), 1–10 (1996)
13. Hormann, K., Labsik, U., Greiner, G.: Remeshing triangulated surfaces with optimal parameterizations. Comput. Aided Des. **33**(11), 779–788 (2001)
14. Frey, P.J.: Generation and adaptation of computational surface meshes from discrete anatomical data. Int. J. Numer. Methods Eng. **60**(6), 1049–1074 (2004)
15. Wang, D., Hassan, O., Morgan, K., Weatherill, N.: Enhanced remeshing from stl files with applications to surface grid generation. Comm. Numer. Methods Engrg. **23**(3), 227–239 (2007)
16. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. ACM Siggraph Comput. Graph. **21**(4), 163–169 (1987)
17. Popov, P., Iliev, V., Fitnev, G.: Quality optimization of seismic-derived surface meshes of geological bodies. In: International Conference on Large-Scale Scientific Computing. pp. 541–551 Springer, (2021)
18. Bonet, J., Peraire, J.: An alternating digital tree (ADT) algorithm for 3d geometric searching and intersection problems. Int. J. Numer. Methods Eng. **31**(1), 1–17 (1991)
19. Thompson, J.F., Soni, B.K., Weatherill, N.P.: Handbook of Grid Generation. (1998). CRC press
20. Weatherill, N., Marchant, M., Hassan, O., Marcum, D.: Grid adaptation using a distribution of sources applied to inviscid compressible flow simulations. Int. J. Numer. Methods Fluids **19**(9), 739–764 (1994)
21. Frey, P.J.: About Surface Remeshing. Citeseer (2000)
22. Borouchaki, H., Laug, P., George, P.-L.: Parametric surface meshing using a combined advancing-front generalized delaunay approach. Int. J. Numer. Methods Eng. **49**(1–2), 233–259 (2000)
23. Lee, C.: On curvature element-size control in metric surface mesh generation. Int. J. Numer. Methods Eng. **50**(4), 787–807 (2001)
24. Wang, D., Hassan, O., Morgan, K., Weatherill, N.: Eqsm: An efficient high quality surface grid generation method based on remeshing. Comput. Methods Appl. Mech. Eng. 195(41), 5621–5633,: John H. Argyris Memorial Issue, Part II (2006)
25. Walton, D.J., Meek, D.S.: A triangular g1 patch from boundary curves. Comput. Aided Des. **28**(2), 113–123 (1996)
26. Frey, P.J., Borouchaki, H.: Geometric surface mesh optimization. Comput. Vis. Sci. **1**(3), 113–121 (1998)
27. Frey, P.J., Borouchaki, H.: Surface mesh quality evaluation. Int. J. Numer. Method. Biomed. Eng. **45**(1), 101–118 (1999)
28. Knupp, P.M.: Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantities. part i-a framework for surface mesh optimization. Int. J. Numer. Methods Eng. **48**(3), 401–420 (2000)
29. Weatherill, N.P., Hassan, O.: Efficient three-dimensional delaunay triangulation with automatic point creation and imposed boundary constraints. Int. J. Numer. Methods Eng. **37**(12), 2005–2039 (1994)