Finite element meshing of three-dimensional faulted domains

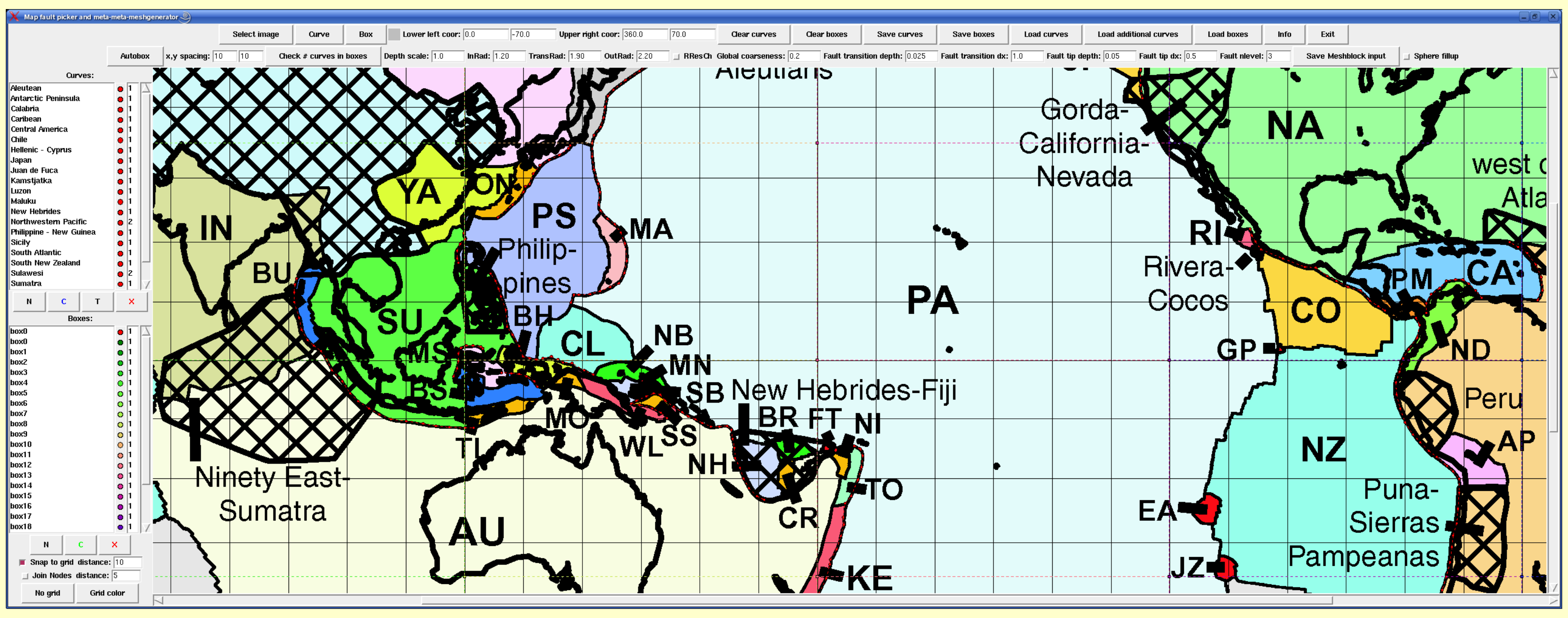
Peter van Thienen and Thomas Geenen. Faculty of Geosciences, Mantle dynamics group, Utrecht University, The Netherlands. thienen@geo.uu.nl, geenen@geo.uu.nl

Abstract

Meshing of 3-D spaces for finite element applications is often a complicated and time-consuming process. The introduction of internal boundaries in order to represent (subduction) faults further complicates the mesh generation process. We present two new tools which result in a significant simplification of the meshing process for geodynamic problems. The first is a graphical tool to facilitate the step from geodynamical maps to finite element meshes (box 1). The second is a mesh generator which uses a regular grid to rapidly generate a finite element mesh for a faulted spherical shell or spherical shell segment (box 1). It includes radial mesh refinement, and iterative optimization of the finite element aspect ratio. Currently in the final stages of development, these tools will be applied to global and regional scale geodynamical modeling studies, using the large-scale mantle convection model of Geenen et al. (see poster "Large scale mantle dynamics modeling").

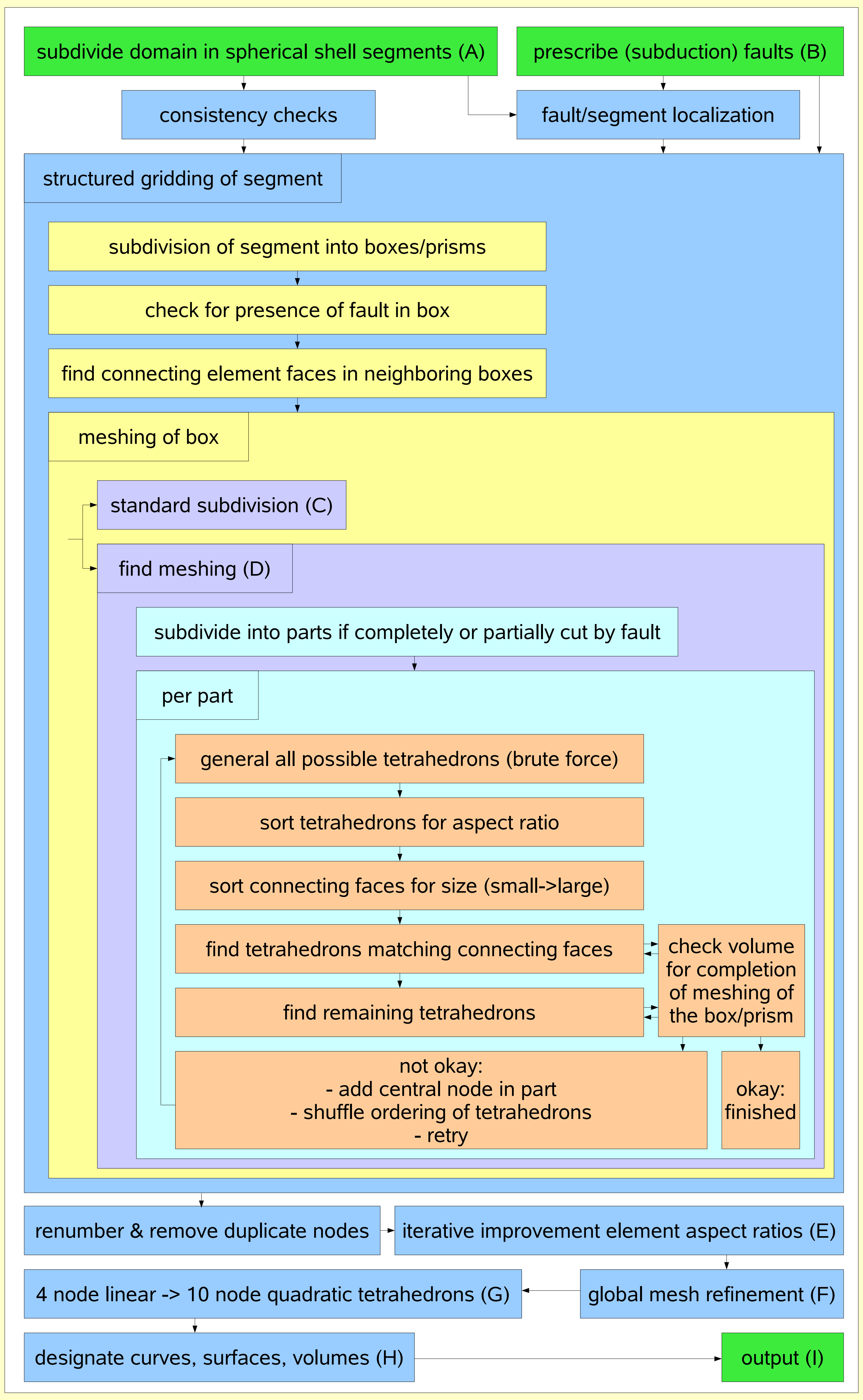
box 1: Geodynamical map meshing tool

Starting from any geodynamical map (the example showing a map from Bird et al., 2003, G3), this tools allows simple manual (subduction) fault picking, importing and exporting of individual faults, and manual and/or automated subdivision of the domain required for the mesh generator (see box 2). In cross-section, subduction faults are anchored to their surface point, an inflection point, and the fault tip, using a spline interpolation to provide an arbitrarily high resolution of the fault parameterization in between. A change in resolution, e.g. at mid-mantle depth, can be included. This tool generates an output file which can be immediately processed by the mesh generator described in box 2.

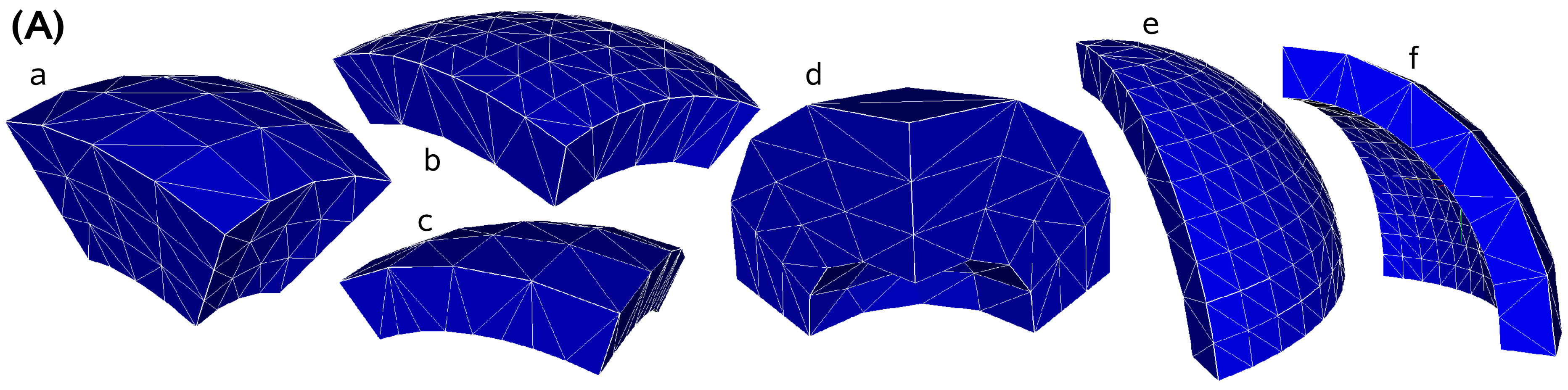


box 2: Mesh generator

The mesh generation process is facilitated by the construction of the domain from a small number of basic building blocks. These building blocks are then subdivided in a regular set of smaller units, which can be meshed relatively easily. The flow diagram below describes the meshing, with several issues being highlighted and explained in more detail below.



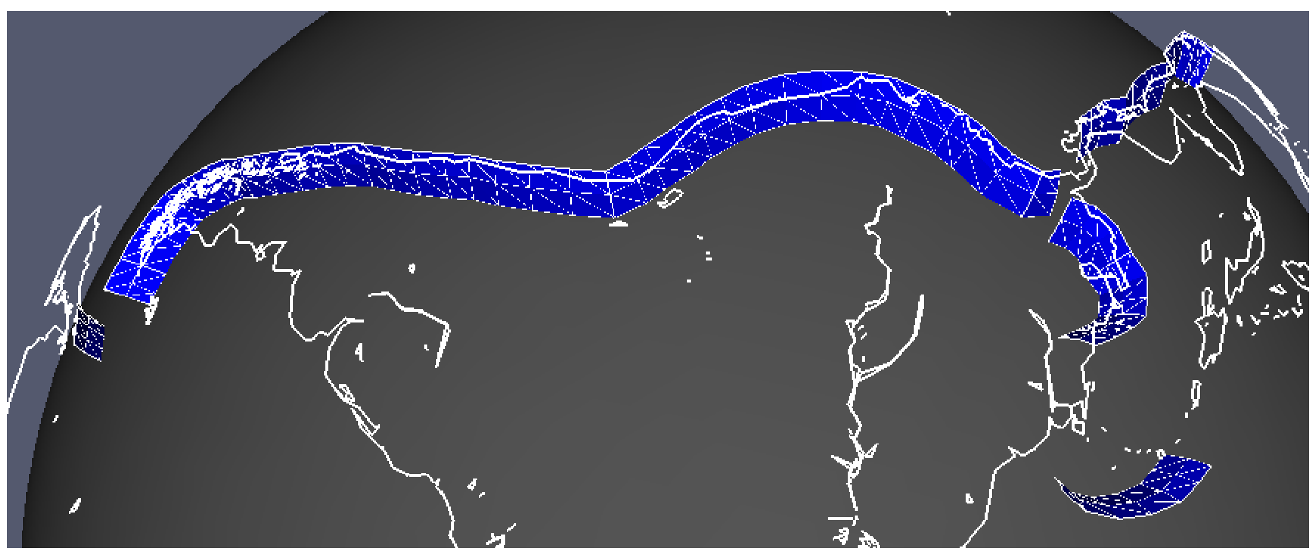
(A)



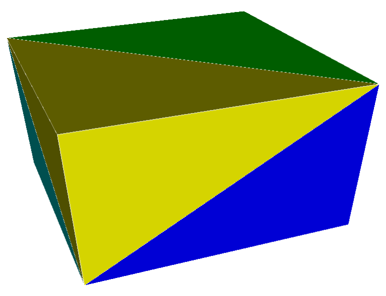
Basic building blocks of the mesh, including pseudo-boxes with uniform (a) and radially changing (b,c) resolution, and pseudo-prisms with uniform (d) and radially changing resolution (e,f)

(B)

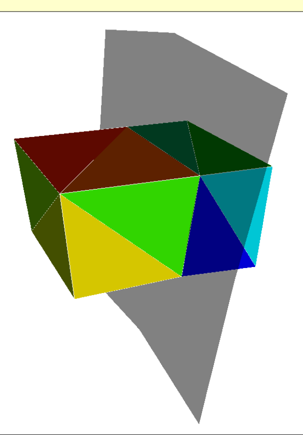
Example of the prescription of subduction faults around South-America. Each fault is included as a triangulated surface with equidistant radial node spacing and a non-uniform lateral distribution.



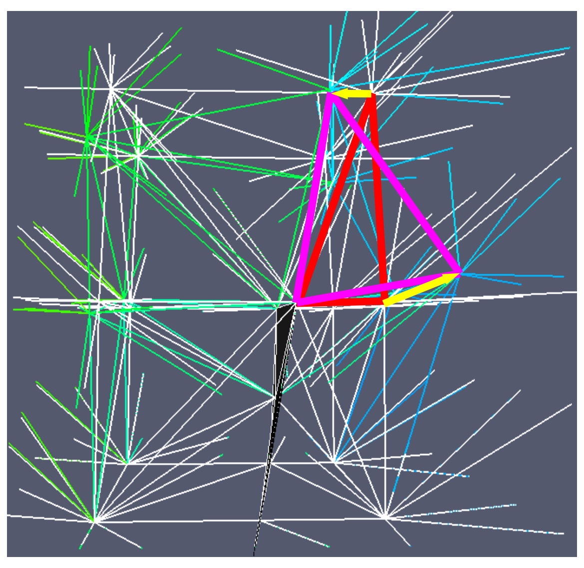
(C)



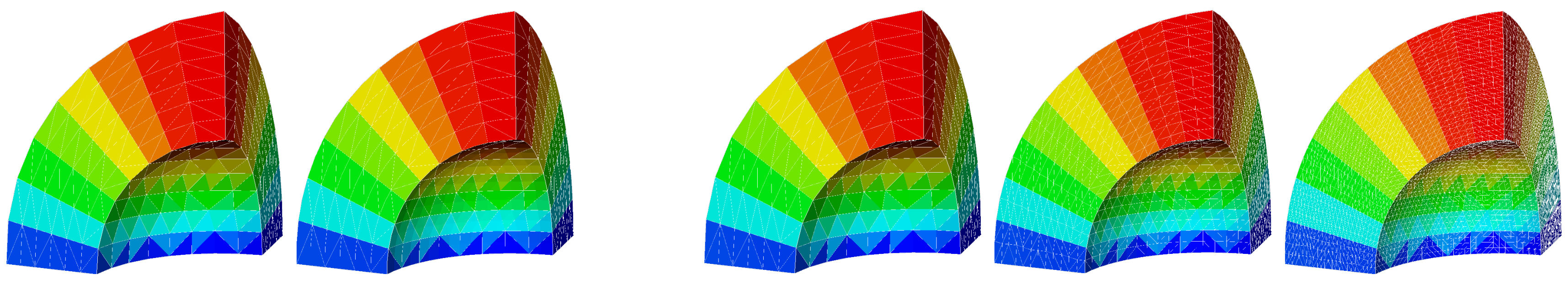
(D)



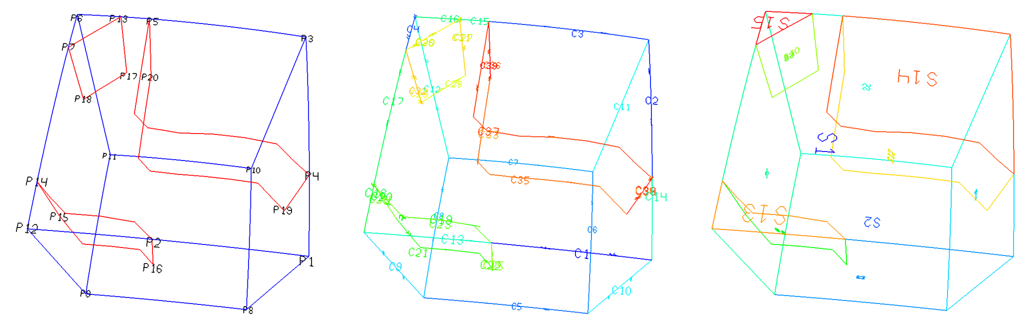
(E) Iterative improvement of the mesh (experimental) consists of optimization of the individual and average element aspect ratio. For each set of opposite edges in each tetrahedron, the distance between the edges is compared, and its deviation from an ideal aspect ratio element results in the prescription of 'force vectors' which either pull the opposite edges towards each other or push them away from each other. These are summed for all edges of all elements, applying an element volume dependent weighing factor. Nodes are moved as a function of the net force (except for boundary and fault nodes), and the procedure is restarted unless convergence has been reached. For relatively simple geometries around faults, this results in a significant improvement of the element aspect ratios. For more complex geometries, the results vary, and more work is required.



(F,G) The midpoints of the edges, if required moved to the same radius as the vertices of the edge, are added as nodal points to increase the order of the elements. Optionally, using these midpoints, each quadratic element can be subdivided into 8 linear elements, which can then undergo the same procedure.



(H) In order to be able to prescribe boundary conditions, the faults and boundaries need to be identified and named in accordance with the requirements of the finite element code which is going to use the mesh (based on the commercial SEPRAN package)



(I) Some examples of finite element meshes produced by the mesh generator and an application. a) high resolution uniformly meshed spherical shell segment (2.26 million quadratic elements, 129 nodes in radial direction); b) meshing of faulted spherical shell segment, with the fault in dark blue, and the fault mesh nodes in brighter blue; c) multiple radial resolution changes; d) snapshot of a numerical convection experiment in a spherical shell segment showing the temperature and flow field (temperature isosurfaces at T=0.1, 0.3, 0.5, 0.7, 0.9).