Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 163 (2016) 46 – 58

Procedia Engineering

www.elsevier.com/locate/procedia

25th International Meshing Roundtable

Mesh scaling for affordable solution verification

Matt Staten[a,*], Brian Carnes[a,*], Corey Mcbride[b], Clint Stimpson[b], Jim Cox[a]

[a] *Sandia National Labs, Albuquerque, NM, USA*
[b] *Elemental Technologies, Inc., American Fork, Utah, USA*

Abstract

Solution verification is the process of verifying the solution of a finite element analysis by performing a series of analyses on meshes of increasing mesh densities, to determine if the solution is converging. Solution verification has historically been too expensive, relying upon refinement templates resulting in an 8X multiplier in the number of elements. For even simple convergence studies, the 8X and 64X meshes must be solved, quickly exhausting computational resources. In this paper, we introduce Mesh Scaling, a new global mesh refinement technique for building series of all-hexahedral meshes for solution verification, without the 8X multiplier. Mesh Scaling reverse engineers the block decomposition of existing all-hexahedral meshes followed by remeshing the block decomposition using the original mesh as the sizing function multiplied by any positive floating number (e.g. 0.5X, 2X, 4X, 6X, etc.), enabling larger series of meshes to be constructed with fewer elements, making solution verification tractable.

Peer-review under responsibility of the organizing committee of IMR 25

Keywords: Mesh Refinement; Solution Verification; Remeshing; Hexahedra

1. Introduction

In order to quantify the numerical error in finite element models, a sequence of meshes is typically required with increasing resolution (reduced element size). The current state-of-the-art approach is to take a base mesh and produce finer meshes using uniform mesh refinement (UMR), under which every element is subdivided according to a refinement pattern. For example a hexahedral element is typically split into eight child elements. In practice, the cost to run the computational model on even a sequence of three meshes (base mesh plus two levels of UMR: 8X, 64X) is prohibitive requiring at least a factor of 100 times the base computational cost, making solution verification with UMR on industrial models infeasible. In addition, UMR cannot be used to coarsen a mesh.

In this paper, we introduce Mesh Scaling, which globally increases or decreases the number of elements in all-hexahedral meshes by any positive floating point multiplier. By allowing arbitrary scale factors, larger series of meshes with fewer elements can be produced significantly reducing the cost of numerical error estimation. As illustrated in Figure 1, Mesh Scaling traverses an existing mesh identifying and propagating constraints from CAD associativity, material assignments, boundary conditions, and mesh irregularities to construct a block decomposition. Each block in the decomposition is either a structured block (Figure 2a) or a swept block (Figure 2b) which can be meshed with

* Corresponding author
E-mail address: mlstate@sandia.gov, bcarnes@sandia.gov

1877-7058
Peer-review under responsibility of the organizing committee of IMR 25
doi:10.1016/j.proeng.2016.11.015

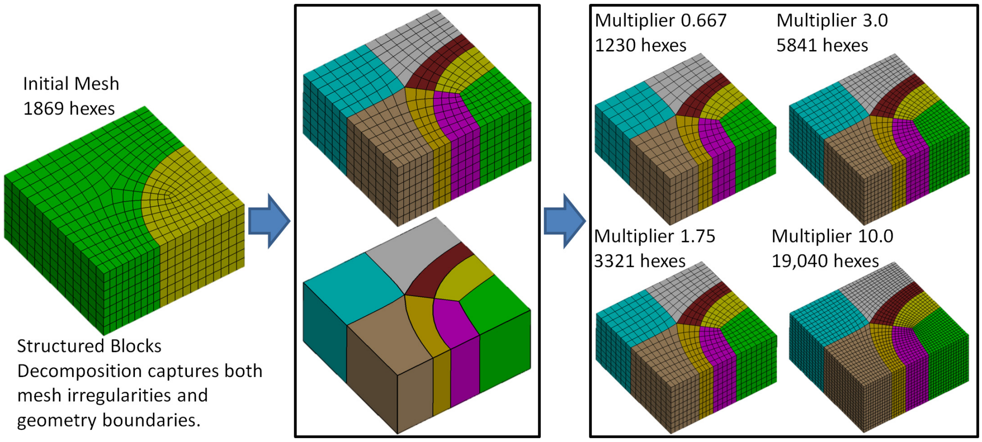

Fig. 1. Overview of Mesh Scaling with a structured block decomposition.

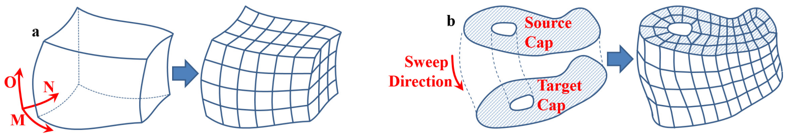

Fig. 2. Block types used in Mesh Scaling. a) a structured block. b) a swept block.

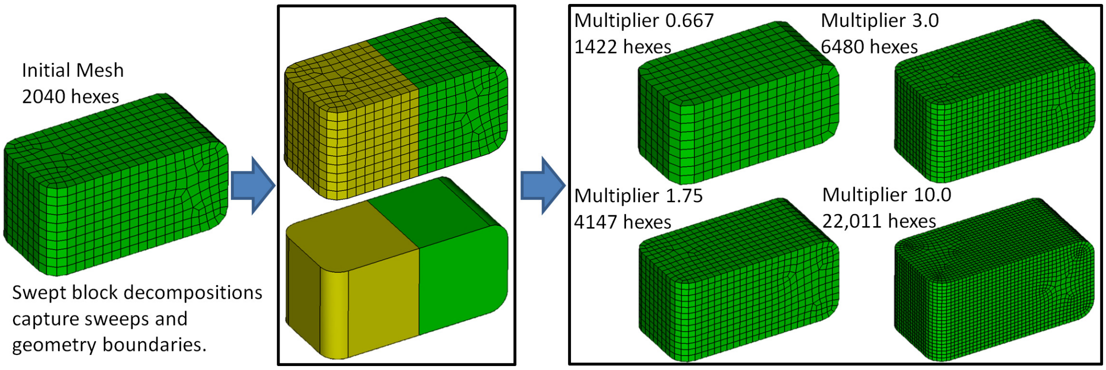

Fig. 3. Overview of Mesh Scaling with a swept block decomposition.

a simple all-hex meshing algorithm. The original mesh is then deleted and the block decomposition is remeshed with a sizing function based on sizes from the original mesh, scaled by an input multiplier.

Figure 1 demonstrates Mesh Scaling with structured blocks only, which provides evenly refined meshes for highly-structured original meshes. However, on other, less-structured, models, structured blocks alone results in a streaky, uneven refinement. Figure 3 demonstrates Mesh Scaling with swept blocks, which provides evenly refined meshes for both structured and unstructured meshes, but changes the number and location of irregular nodes (i.e. nodes with a non-optimal valence [12]).

In some cases, a tool such as Mesh Scaling can be avoided by simply re-applying the original recipe (script) for generating the mesh at different sizes. However, this is not always possible. The recipe may no longer be available. Even, if the recipe is available, running it at a different size is often unsuccessful because of the propagating constraints in hexahedral meshes. Also, re-applying the recipe requires access to the original CAD, and the original CAD decomposition upon which the recipe is built, which may not be available either. In addition, mesh modifications such as smoothing, refinement, pillowing, etc. may have been done to the mesh after the initial generation and these would not be re-applied by running the recipe. Finally, the mesh generation recipe most likely does not include application