8th. World Congress on Computational Mechanics (WCCM8)
5th European Congress on Computational Methods in Applied Sciences and Engineeering (ECCOMAS 2008)
June 30 –July 5, 2008
Venice, Italy

HIERARCHICAL GRID ADAPTATION FOR HYBRID MESHES

Matthias Möller[1]

[1] Institute of Applied Mathematics, Technische Universität Dortmund
Vogelpothsweg 87, D-44227 Dortmund, Germany
matthias.moeller@math.uni-dortmund.de

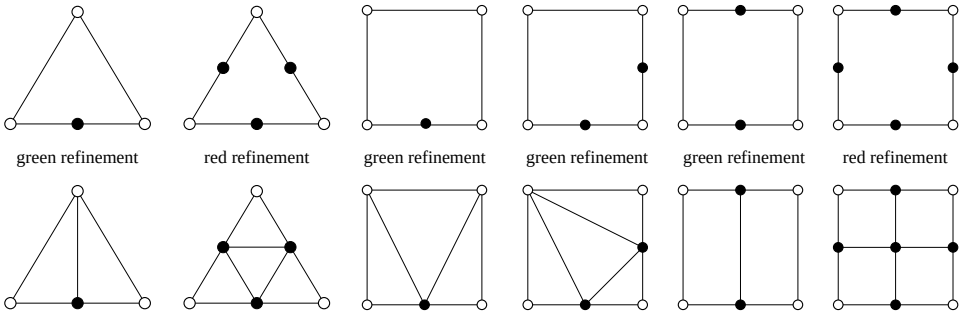
Key Words: *Grid Adaptation, Data Structures, High-Resolution Schemes, Finite Element Method.*

ABSTRACT

Grid adaptation based on the red-green strategy [1] is revisited. A new approach to the construction of conforming hybrid triangulations with linear time and space requirement is presented. The main advantage of the method is that mesh refinement and re-coarsening yields a sequence of hierarchical grids that can be used within multigrid schemes. All information are reconstructed from one nodal vector storing the 'date of birth' for each vertex so that no tree-based data structures are required.

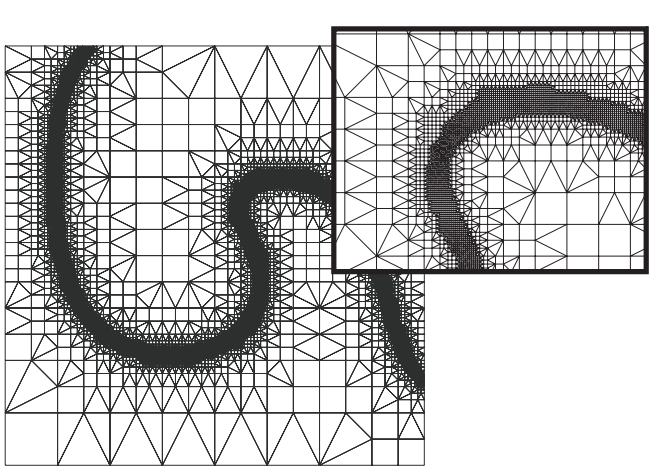
Starting from an initial grid $\mathcal{T}_0 = (\mathcal{E}_0, \mathcal{V}_0)$ which consists of elements $\mathcal{E}_0 = \{\Omega_k, k = 1, \ldots, N_{\mathrm{E}}\}$ and comprises vertices $\mathcal{V}_0 = \{\mathrm{v}_i : i = 1, \ldots, N_{\mathrm{V}}\}$ new cells are inserted step-by-step so as to locally adapt the computational grid to the estimated solution error. The task of the adaptation procedure is to transform a given triangulation $\mathcal{T}_m$ into a new grid $\mathcal{T}_{m+1}$ that is also conforming. The mesh can be updated at fixed time intervals or based on some (global) error estimator/indicator. The entire grid *genealogy* of the sequence $\{\mathcal{T}_m\}_{m=0}^{M}$ is recovered from the *generation function* $g : \mathcal{V}_m \rightarrow \mathbb{N}_0$ [2]. By definition, all nodes of the initial triangulation belong to generation zero. If a new vertex is inserted at the midpoint of an edge its generation index is determined as the maximum generation of the two endpoints increased by one [2]. For a new vertex inserted at the center of a quadrilateral, the maximum is taken over the generation indices of all surrounding nodes and increased by one [3]. It is trivial to prescribe a maximum number of refinement levels so as to control the computational costs of dynamic grid adaptation for time-dependent problems. Of course, the vertex with largest generation index is always the one which was inserted most recently and can be removed foremost.

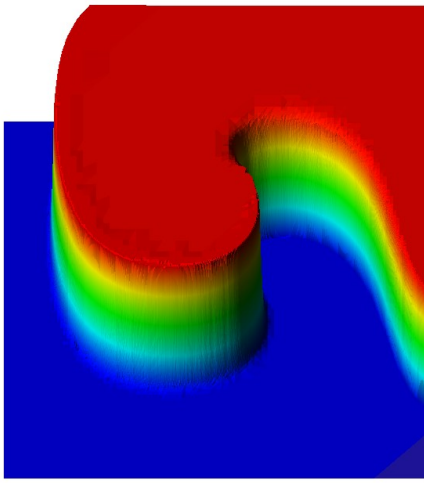
In two space dimensions, the classical red/green refinement patterns are as follows, whereby green elements need to be converted into the corresponding red ones prior to performing further subdivision.



For time-dependent flow simulations, the mesh is locally re-coarsened in regions where the error becomes sufficiently small. The removal of elements/vertices is regarded as a reverting process 'inverse' to subdivision. It is based on the design principle that any refinement operation applied in one step, e.g. $\mathcal{T}_m \rightarrow \mathcal{T}_{m+1}$, can be 'undone' in a single step $\mathcal{T}_{m+1} \rightarrow \mathcal{T}_{m+2}$ so that $\mathcal{T}_m$ and $\mathcal{T}_{m+2}$ are topologically equivalent. To this end, a generalization of the generation function g is introduced $l : \mathcal{V}_m \rightarrow \mathbb{Z}$, whereby the absolute values of both quantities coincide. Depending on the sign of $l(\mathrm{v}_p)$, node v_p can be removed from the triangulation if $l(\mathrm{v}_p) > 0$ and it has to be kept otherwise. Of course, all vertices of the initial mesh belong to generation zero and are strictly preserved by construction. All other nodes $\mathrm{v}_p \in \mathcal{V}_m \setminus \mathcal{V}_0$ are initialized as $l(\mathrm{v}_p) := g(\mathrm{v}_p)$ and 'locked' gradually by a two-step algorithm [2] that terminates in linear time $\mathcal{O}(|\mathcal{E}_m|)$. The locking of nodes is performed after the identification of elements that need to be refined due to accuracy reasons and before the actual refinement process.

The characterization of green and red triangles/quadrilaterals as well as their relation to neighboring cells and macro elements is solely based on the generation function [2,3]. It can be stored in linear space $\mathcal{O}(|\mathcal{V}_m|)$ so that all look-up operations can be performed in constant time $\mathcal{O}(1)$ and no additional tree-based data structure is required. In order to efficiently identify groups of elements resulting from the same refinement operation, a localized variant of the two-level ordering strategy by Turek [4] is adopted. In essence, the numbering of local degrees of freedom within each element strictly follows a set of rules which are defined *a priori*. Thus, matching edge/face neighbors can be directly addressed without looping over all neighboring cells. The marking of elements for refinement can be efficiently performed in linear time and space $\mathcal{O}(|\mathcal{E}_m|)$. It is worthwhile to adopt binary arithmetic so that the pending bisection of edges/faces can be indicated by flipping the corresponding bit in MSB representation. A similar strategy can be applied to convert nodal generation indices to *element states*.





REFERENCES

[1] R.E. Bank, A.H. Sherman and A. Weiser. "Refinement algorithms and data structures for regular local mesh refinements". in: *Scientific Computing*, eds. R. Stepleman et al. (IMACS, North-Holland, Amsterdam, 1983) pp. 3–17.

[2] D. Hempel. "Rekonstruktionsverfahren auf unstrukturierten Gittern zur numerischen Simulation von Erhaltungsprinzipien". PhD-thesis, University of Hamburg, 1999.

[3] M. Möller. "Adaptive High-Resolution Finite Element Schemes". PhD-thesis, in prep.

[4] S. Turek. "On ordering strategies in a multigrid algorithm". in: *Notes on Numerical Fluids Mechanics*, volume 41 (Proc. 8th GAMM-Seminar, Vieweg, Kiel, 1992)