Dynamically Loadable, Selectively Refinable Progressive Meshes, Subdivision Quarks, and All That

S. Seeger, G. Häusler

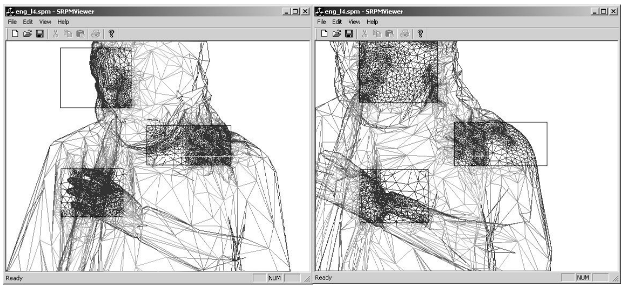

Figure 1: While moving the model the regions of interest change. The corresponding refinement and coarsening are performed in real-time.

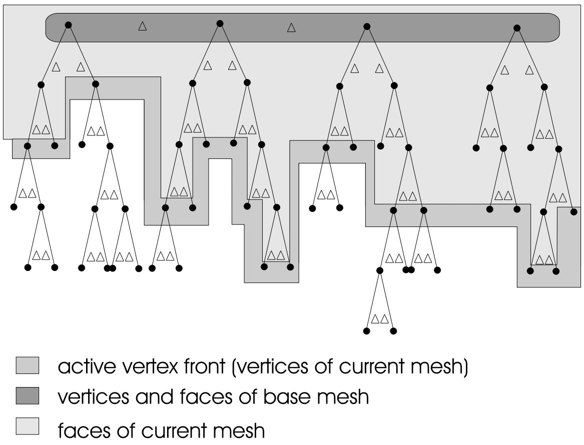

Figure 2: The data structure of a Selectively Refinable Progressive Mesh corresponds to a vertex forest.

In [1] we introduced the concept of Selectively Refinable Progressive Meshes (SRPM) as the ideal data structure for complex numerical tasks. Meanwhile we finished the implementation of this data structure, provided a viewer for the visualization of the SRPM (Fig. 1), and extended the idea of SRPMs in two ways: in dynamically loading the mesh we enable to work with arbitrarily large models, and in introducing the concept of subdivision quarks [2] we allow to infinitely refine selected regions of interest (ROI).

SRPMs are based on Progressive Meshes (PM). In the PM data structure a usual triangle mesh is represented by a small base mesh plus an ordered set of vertex split operations resulting in a continuous hierarchy of triangle meshes. Since a vertex split is a local operation it is possible to skip some of the split operations and allowing in this way to selectively refine the mesh to arbitrary ROI. In order to ensure that the same vertex splits always result in the same mesh independent of the order in which splits and collapses are performed (i.e. in order to ensure uniqueness of the mesh) the split vertex is reproduced during the split operation resulting in the SRPM data structure of a vertex forest shown in Fig. 2. Each selectively refined mesh corresponds to a so called *active vertex front* through this vertex hierarchy.

With the concepts described in the original work to SRPMs [3] the whole SRPM data structure has to be built from a PM sequence once during initialization. While this may take a long time for medium-sized to large models, such a concept totally prohibits the use of really huge models with several billions of triangles, since such models neither fit into main nor into virtual memory. In addition, the advantage of PMs to allow progressive transmission of the mesh over low-bandwidth communication lines like the internet is getting lost in such an SRPM. Therefore we extended the SRPM data structure by enabling to load only that parts of the PM sequence that are needed for the current ROI. In principle, this dynamic loading enables to work with arbitrarily large models.

Soon or later there is no split operation any more on disk for a given ROI, so that we cannot further refine the mesh in this region. However, it is possible to invent some split operations. Although we have complete freedom in choosing such split operations, we tried to perform splits that are consistent with a usual subdivision scheme as in [4]. In this way we know from subdivision theory that the limit surface that results after infinitely many subdivision steps will be e.g. C^1-continuous in the case of [4]. In addition we have a strong relationship to wavelet theory. Since we need 3 split operations to reproduce one usual elementary subdivision step and this is similar to the way physicists use quarks for describing elementary particles like protons and neutrons, we called these devised split operations *subdivision quarks.*

[1] S. Seeger, G. Häusler, Selectively Refinable Progressive Meshes: The ideal data structure for complex numerical tasks?, *Lehr. f. Optik, ann. rep. 2000*, p. 16, Erlangen.
[2] S. Seeger, K. Hormann, G. Häusler, G. Greiner, A Sub-Atomic Subdivision Approach. In *Vision, Modeling and Visualization '01*, pp. 77-85, Stuttgart, 2001.
[3] H. Hoppe, View-dependent Refinement of Progressive Meshes, *Siggraph '97 Proceedings*, pp. 189-198, 1997.
[4] N. Dyn et al., A Butterfly Subdivision Scheme for Surface Interpolation With Tension Control, *ACM Transactions on Graphics*, vol. 9, 1990, pp. 160–169.