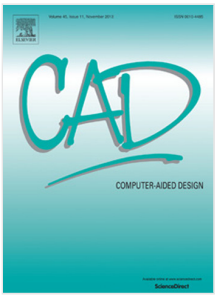
Accepted Manuscript



A two-level multithreaded Delaunay kernel

Jean-François Remaclea

PII: S0010-4485(16)30093-8
DOI: http://dx.doi.org/10.1016/j.cad.2016.07.018
Reference: JCAD 2463

To appear in: *Computer-Aided Design*

Please cite this article as: Remaclea J-F. A two-level multithreaded Delaunay kernel. *Computer-Aided Design* (2016), http://dx.doi.org/10.1016/j.cad.2016.07.018

A Two-Level Multithreaded Delaunay Kernel

Jean-François Remacle[a]

[a] *Université catholique de Louvain, Institute of Mechanics, Materials and Civil Engineering (iMMC), Bâtiment Euler, Avenue Georges Lemaître 4, 1348 Louvain-la-Neuve, Belgium*

Abstract

This paper presents a fine grain parallel version of the 3D Delaunay Kernel procedure using the OpenMP (Open Multi-Processing) API. A set $S = \{p_1, \ldots, p_n\}$ of n points is taken as input. S is initially sorted along a space-filling curve so that two points that are close in the insertion order are also close geometrically. The sorted set of point is then divided into M subsets S_i, $1 \leq i \leq M$ of equal size n/M. The multithreaded version of the Delaunay kernel inserts M points at a time in the triangulation. OpenMP barriers provide the required synchronization that is needed after each multiple insertion in order to avoid data races. This simple approach exhibits two standard problems of parallel computing: load imbalance and parallel overheads. Those two issues are addressed using a two-level version of the multithreaded Delaunay kernel. Tests show that triangulations of about a billion tetrahedra can be generated on a 32 core machine (Intel Xeon E5-4610 v2 @ 2.30GHz with with 128 GB of memory) in less that 3 minutes of wall clock time, with a speedup of 18 compared to the single-threaded implementation.

Keywords: Delaunay triangulation, parallel computing, OpenMP

1. Introduction

In the last decades, the size of the finite element meshes that are used in industry for scientific computing has grown considerably due to the availability of massively parallel computers. It is nowadays not uncommon to generate

*Jean-Franois Remacle.