CujaMara: Restriction-Aware Block Splitting for Optimised Load Balance

Gregor Matura[a], Martin Siggel[a], Achim Basermann[a]

[a] *Distributed Systems and Component Software, Simulation and Software Technology, German Aerospace Center, Linder Höhe, 51147 Köln, Germany*

Abstract

Every TRACE simulation is based on a geometry file containing the model that will be simulated, for example, of a turbomachinery. This model consists of blocks, each block being a distinct component of the turbomachinery and thus varying strongly in complexity. As the number of components, or blocks, additionally varies from geometry to geometry the parallel processing of the blocks becomes complicated. A naive distribution of the blocks generally leads to a poor load balance and thus a bad parallel efficiency.
Splits of the blocks into smaller chunks can improve the situation tremendously, but we have to act with caution. Every split of a block improves the ability to distribute all blocks more evenly, but increases total communication costs at the same time. Thus, a vast amount of splits, that, in turn, ensures a well-balanced distribution, is pointless; a minimal number of splits is required. Additionally, some blocks are restricted by their boundaries: Splits in certain directions are forbidden.
In this talk, we present CujaMara, a stand-alone tool performing a minimal number of splits optimised for a subsequent distribution of the blocks to a specified number of processes and in compliance with all given split restrictions. CujaMara does a heuristic search for the splits and aims at an optimal load balance for the targeted number of processes. It considers various costs to compute this load, e.g., computational costs per cell and communication costs of cells at block surfaces. The output is a geometry file where the calculated minimal number of optimal splits is applied, and all initial split restrictions are preserved.
We demonstrate the capabilities of our tool with different geometries (compare figures 1 and 2 for two examples) for up to a hundred of target processors. Here, CujaMara achieves almost optimal load balance for reasonably chosen numbers of processors (dependent on the complexity of the geometry). CujaMara is highly configurable to best possibly meet the needs of any given geometry and simulation. CujaMara's run time depends on the initial configuration, i.e., the number of initial blocks and their individual restrictions; it usually takes only seconds. For cost and energy efficient resource allocation, CujaMara warns if the optimal number of processes falls below the specified number.

Email addresses: `gregor.matura@dlr.de` (Gregor Matura), `martin.siggel@dlr.de` (Martin Siggel), `achim.basermann@dlr.de` (Achim Basermann)

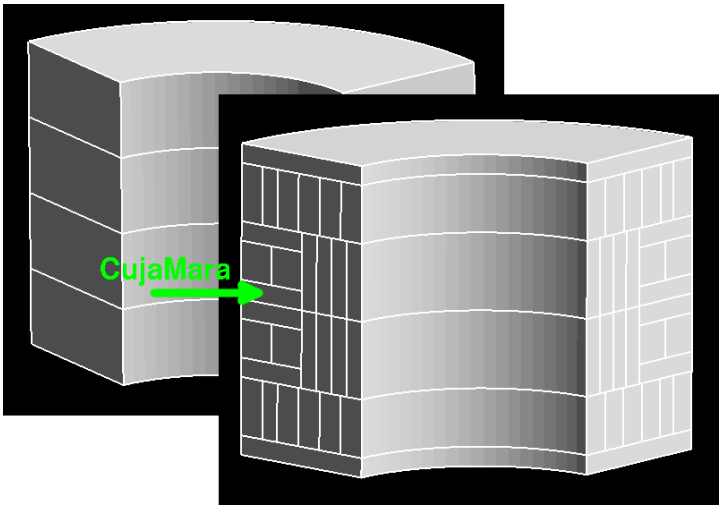

Figure 1: CujaMara applied on a simple block geometry, number of target processors: 32

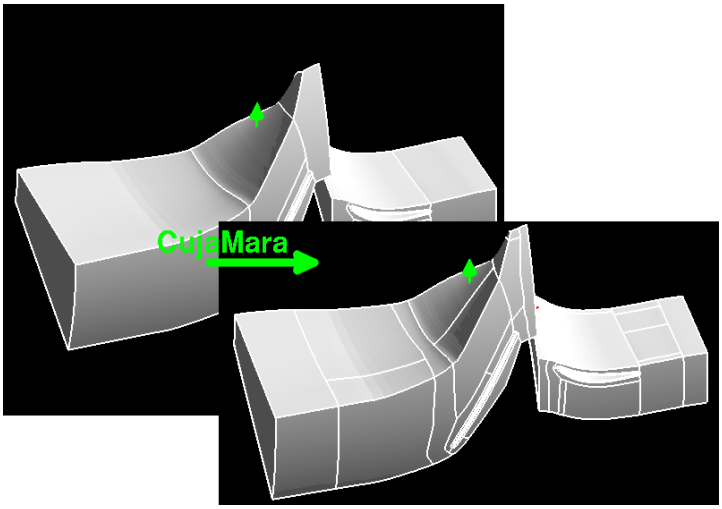

Figure 2: CujaMara applied on a advanced block geometry, number of target processors: 32