Abstract Ph.D. Thesis

»Shape Optimized Graph Partitioning«

In this thesis we address the graph partitioning and repartitioning problem. In particular, we focus on load balancing in a distributed numerical simulation. We provide an overview on commonly applied techniques and list state-of-the-art implementations.
Furthermore, we present several enhancements to existing approaches. We introduce the Graph-Filling Curves, present a better evaluation scheme, and improve the Helpful-Set bisection refinement procedure that is implemented in the Party library. Additionally, we summarize some new results on diffusion schemes for certain network types.
The main contribution of this thesis is the new repartitioning heuristic Flux. In contrast to other existing libraries, we follow an alternative strategy and do not focus on a small edge-cut. Instead, we optimize the domain shapes by applying a diffusion process embedded inside a learning framework. We formulate and analyse the new diffusion scheme FOS/C that meets our special requirements.
While the solutions computed with Flux also contain a small number of cut edges, our experiments show that the main benefit of the new heuristic is the ability to find well shaped and connected domains with a small number of boundary vertices, what meets the requirements of distributed numerical simulations. Furthermore, the proposed mechanism contains many calculations that can easily be performed in parallel. The main drawback of Flux is the large number of involved numerical computations. We present some techniques that reduce the computational load.
Nevertheless, in its current implementation the Flux library is two to three magnitudes slower than other distributed state-of-the-art heuristics. Depending on the application, this overhead might or might not be justified by the better solution quality. Since some of the algorithms in the PadFEM simulation environment depend on good partition shapes and require connected domains, the Flux library is currently the only applicable load balancer.