A parallelized interval arithmetics based reliable computing method on GPU

Zsolt Bagóczki, Balázs Bánhelyi

Videocards has outgrew in the past to be only a simple tool for graphic display. With their high speed video memories, lots of math units and parallelism, they provide a very powerul platform for general purpose computing tasks, in which we have to deal with large datasets, which are highly parallelizable, have high arithmetic intensity, etc. Our selected platform for testing is the CUDA (Compute Unified Device Architecture), that grants us direct reach to the virtual instruction set of the video card, and we are able to run our computations on dedicated computing kernels. The CUDA development kit comes with a useful toolbox with a wide range of GPU based function libraries.

In this parallel environment we implemented a reliable method. Hence the finite precision of computers, for achieving a reliable method, we needed to use interval arithmetics, so instead of giving a rounded, inaccurate result, we are able to give an interval, that is certainly including the exact answer. Our method is based on the branch-and-bound algorithm, with the purpose to decide whether or not any given property applies for a two-dimensional interval. This algorithm will give us the opportunity to use node level parallelization. Node level parallelization means, that our nodes are evaluated simultaneously on multiple threads we start on the GPU. This is called low-level, or type 1 parallelization, since we do not modify the searching trajectories, neither do we modify the dimensions of the branch-and-bound tree [1].

For testing, we choose the circle covering problem. It is the dual of circle packing, where our goal is to find the densest packing of a given number of congruent circles with disjoint interiors in a unit square. In circle covering, we aim for finding the full covering of the unit square with congruent circles of minimal radii. Overlapping interiors are allowed. For achieving easily scalable test cases, we will decide the covering with precalculated circles, where the diameter of the circles is the diagonal of the unit square, divided by a given number, n, and the number of the circles is the square of n. We scaled the problem up to three dimensions, and ran tests with sphere covering problems, too, and we discuss the possibility of scaling the problem up to any dimensions easily.

References

[1] Teodor Gabriel Crainic, Bertrand Le Cun, and Catherine Roucairol: Chapter 1. Parallel Branch-and-Bound Algorithms, Parallel Combinatorial Optimization. Published Online (2006)