Extreme performance Model Predictive Control using a GPGPU

Isak Nielsen[*], Daniel Axehill[†]

November 19, 2013

This MSc thesis is a challenging task which is an opportunity to work at the research front on a hot topic, developing the next generation of solvers for model predictive control problems!

Model predictive control (MPC) is one of the most popular control strategies of today. This type of control can be applied to linear as well as non-linear systems. The fundamental idea in MPC is to solve an optimization problem on-line in each sample which depends on the measured or estimated state of the plant. The optimization is performed over the prediction horizon N and the optimization problem is commonly a quadratic programming (QP). The computational effort grows with increased prediction horizon N, and the computational complexity of state of the art serial solvers scale linearly in N. As more applications with very long prediction horizons arise, this is a bottleneck in achieving high performance control using MPC:s. In ongoing research much effort is spend on the next generation of model predictive control using solvers tailored for parallel hardware. The algorithms that are designed to be run on parallel architectures manage to get logarithmic growth in the prediction horizon, i.e. $\mathcal{O}(\log N)$, which is significantly less for problems with large N.

In order to extend the applicability of MPC, we would like to investigate the possibilities from parallelism. In particular the gains of using a newly developed parallel algorithm for solving the optimization problem should be investigated. The algorithm divides the original optimization problem into several smaller problems that are simpler to solve. It adopts different levels of parallelism by changing the sizes of the subproblem. Since the smaller subproblems all have the same structure, we think that an implementation on a general purpose graphical processing unit (GPGPU) would be efficient. The main objectives in this work are

1. Develop a parallel framework for solving MPC problems on a GPGPU.
   - Review which typical parallel algorithms that are used today
   - Implement the proposed algorithm on a GPGPU
   - Investigate which level of parallelism that is most efficient
2. Test the framework on computationally demanding MPC problems with long prediction horizon, e.g. planning problems in autonomous vehicles.

[*] Tel: 013-282804, E-mail: isak.nielsen@liu.se

[†] Tel: 013-284042, E-mail: daniel@isy.liu.se

The project can be tailored for the student's experience and interests. The expected outcome is a working parallel solver for MPC implemented on a GPGPU. Furthermore, one or more challenging examples where the software is used for high performance MPC should be investigated and used as examples of the performance of the code.

To be able to smoothly solve this task it is preferable that the person has experience of parallel programming (especially for GPGPU:s) and is aware of different implementation techniques. Since the proposed algorithm is still under research, it is important that the student is capable of working individually and assimilating theoretical results.