FPGA-MPC: a FPGA-accelerated QP Solver for Robot Model Predictive Control

Alexander Du (asd2192), Apurva Reddy (akr2177), Roy Hwang (rjh2173), Godwill Agbehonou (gea2118)
March 14, 2025

Introduction

Model Predictive Control (MPC) is a feedback control strategy that has seen great success in a wide variety of robotic applications [1, 2]. Most implementations of MPC leverage trajectory optimization (TO) to that optimize an objective function subject to system dynamics and other constraints. Given their highly nonlinear nature, these problems demand significant computation to achieve optimality. In many cases, problem instances with hundreds/thousands of variables must be solved in milliseconds to achieve real-time performance. As such, careful approximations and simplifications of the underlying optimal control problem can be used to enable real-time edge deployments on a variety of different hardware platforms [3].

Commonly, the slowest part of trajectory optimization is solving a quadratic programming (QP) problem. In this project, we target this bottleneck by developing a low-latency QP solver that exploits parallelism on a FPGA. By reducing computation time, we hope to enable faster real-time control of robotic systems. We are inspired by recent works that leverage hardware acceleration on GPUs [4, 5, 6, 7], as well as aformationed simplifications to enable performant use on resource-constrained microcontrollers [8].

Initial Plan

We plan to first study some related works that accelerate MPC on FPGAs, and propose the following milestones:

- v0.8.0 — Proof of concept: a minimal version where the QP solver is used for MPC of a simple simulated control system like a pendulum or a double integrator.
- v1.0.0 — Integrating the MPC in a closed control loop simulation with an external computer (via Ethernet) that acts as the control system. This is to demonstrate that the system receives sensor data, can produce actions, and acts in real-time.
- v1.2.0: A hardware demonstration that connects the FPGA-accelerated MPC directly to an RC car or robotic manipulator to perform a task like tracking a reference trajectory.

Details

The hardware accelerator will be responsible for solving a QP, perhaps through a linear system that is formed from the QP. The software will interface with the robot control system (recieving current state and sending actions), compute dynamics and cost gradients to form the problem, and orchestrate other methods needed to run online MPC.

There are many algorithms with which to solve QPs: gradient projection, interior point, and active set method. Gradient Projection Methods work best for problems with few and simple constraints. Interior Point Methods are best for problems with many constraints. Active Set Methods work best for problems with few constraints but constraints that update frequently. Our first step is choosing which formulation to optimize. We must also settle on a specific robot instance to demonstrate our solver on. The hardest task will be transcribing the chosen QP Solver algorithm into hardware. Ultimately, in addition to a hardware demo, we plan to benchmark our solver in comparison with an equivalent CPU implementation, and show that it achieves superior performance.

Given the complexity of the full MPC system, to facilitate development, we will use popular simulation engines (such as MuJoCo/PyBullet), dynamics libraries (Pinocchio), and reference open-source software (OSQP, Acados, Casadi) to develop our solver. Also, we are lucky to have access to robotic platforms through the Columbia University Robotics Club and the Barnard Accesible and Accelerated Robotics lab.

C Implementation:

- The robot control system interface will communicate using an appropriate protocol like UART or Ethernet and will read the robot's current state (position, velocity, sensor data)
- Use DMA (Direct Memory Access) to transfer the problem data to teh FPGA and retrieve results.
- It will be important to synchronize computation with the hardware accelerator using interrupts or polling.
- Compute system dynamics and formulate the control objectives and constraints.
- Utilize matrix operations to define the QP structure.

FPGA Specfic Implementation:

- The main idea is using FPGA parallelism to accelerate matrix multiplications. Decompositions, and iterative methods.
- Utilize DSP blocks and Block RAM on the FPGA for efficient arithmetic operations and storage.
- Use DMA (direct memory access) for efficient data transfer of QP matrices and results.

References

[1] F.R.Hogan,E.R.Grau,andA.Rodriguez,“Reactiveplanarmanipula- tion with convex hybrid mpc,” in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 247–253.

[2] P. M. Wensing, M. Posa, Y. Hu, A. Escande, N. Mansard, and A. Del Prete, “Optimization-based control for dynamic legged robots,” IEEE Transactions on Robotics, 2023.

[3] S. Kuindersma, R. Deits, M. Fallon, A. Valenzuela, H. Dai, F. Per- menter, T. Koolen, P. Marion, and R. Tedrake, “Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot,” Autonomous robots, vol. 40, pp. 429–455, 2016.

[4] T. Antony and M. J. Grant, “Rapid Indirect Trajectory Optimization on Highly Parallel Computing Architectures,” vol. 54, no. 5, pp. 1081– 1091.

[5] B. Plancher, S. M. Neuman, R. Ghosal, S. Kuindersma, and V. J. Reddi, “GRiD: GPU-Accelerated Rigid Body Dynamics with Analytical Gradients,” in 2022 International Conference on Robotics and Automation (ICRA). IEEE, pp. 6253–6260.

[6] Y. Lee, M. Cho, and K.-S. Kim, “Gpu-parallelized iterative lqr with input constraints for fast collision avoidance of autonomous vehicles,” in 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2022, pp. 4797–4804.

[7] E. Adabag, M. Atal, W. Gerard, and B. Plancher, “Mpcgpu: Real-time nonlinear model predictive control through preconditioned conjugate gradient on the gpu,” in 2024 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2024, pp. 9787–9794.

[8] K. Nguyen, S. Schoedel, A. Alavilli, B. Plancher, and Z. Manchester, “Tinympc: Model-predictive control on resource-constrained micro- controllers,” in 2024 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2024, pp. 1–7.