# MPCGPU: Real-Time Nonlinear Model Predictive Control through Preconditioned Conjugate Gradient on the GPU

Emre Adabag[1], Miloni Atal*[1], William Gerard*[1], Brian Plancher[2]

1: School of Engineering and Applied Science, Columbia University 2: Barnard College, Columbia University

## The Big Picture:

In this work, we introduce MPCGPU, a GPU-accelerated, **real-time NMPC solver that leverages a preconditioned conjugate gradient (PCG) linear system solver** at its core. We show that MPCGPU increases the scalability and real-time performance of NMPC, solving larger problems, at faster rates. In particular, for tracking tasks using the Kuka IIWA manipulator, **MPCGPU scales to kilohertz control rates** with trajectories as long as **512 knot points**. This is driven by a custom **PCG solver which outperforms CPU-based**, state-of-the-art, linear system solvers by **at least 10x for a majority of solves and 3.6x on average**.

ICRA2024
May 13-17, Yokohama, Japan

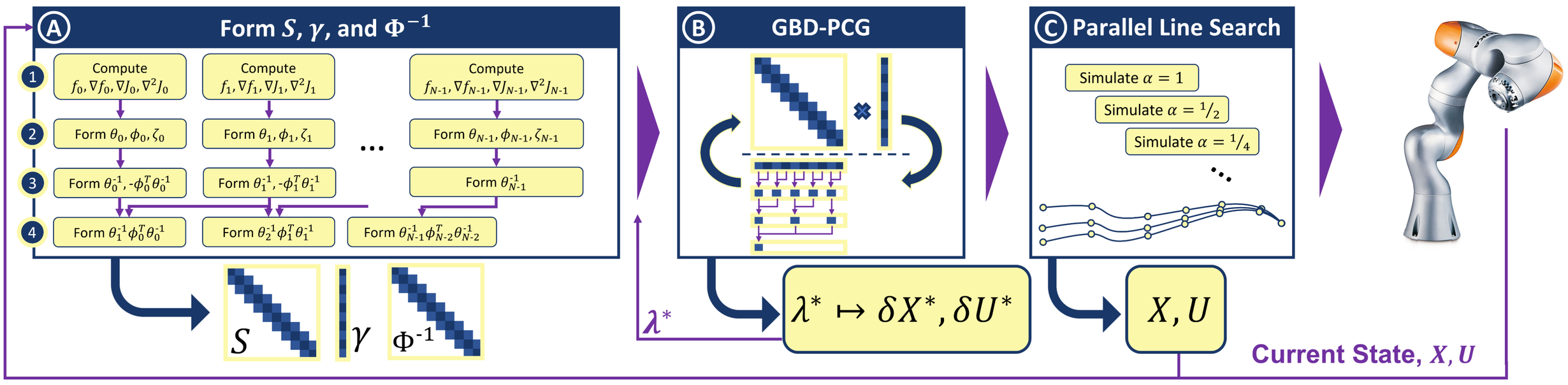
The Barnard Accessible and Accelerated Robotics Lab

## Algorithmic Approach:

MPCGPU solves the NMPC problem through three-steps:

1) On the GPU it **computes S, γ, and $\Phi^{-1}$ in parallel**,
2) Uses the **GPU-Accelerated Block-Diagonal PCG algorithm (GBD-PCG) to compute λ*** and reconstruct δX*, δU* efficiently by re-factoring the PCG algorithm to better expose parallelism and exploit the sparsity in the trajectory optimization problem,
3) Uses a **parallel line search** to form the final trajectory.

This trajectory is passed to the (simulated) robot and the current state of the (simulated) robot is measured and fed back into our solver which is run again, **warm-started with our last solution**.



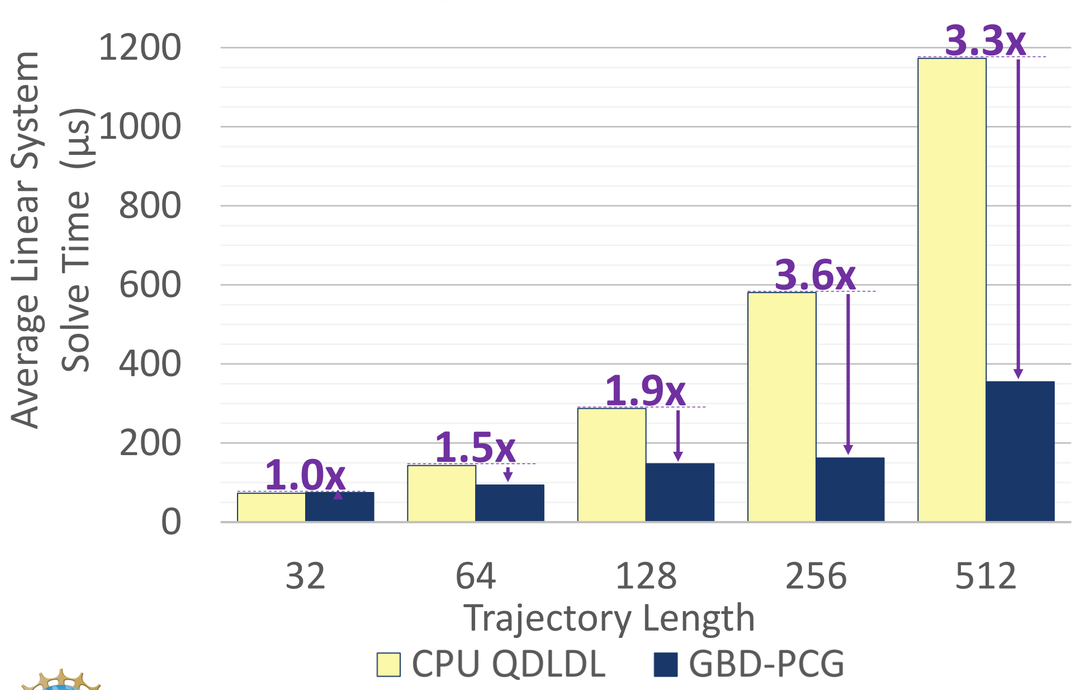

## Experimental Results:

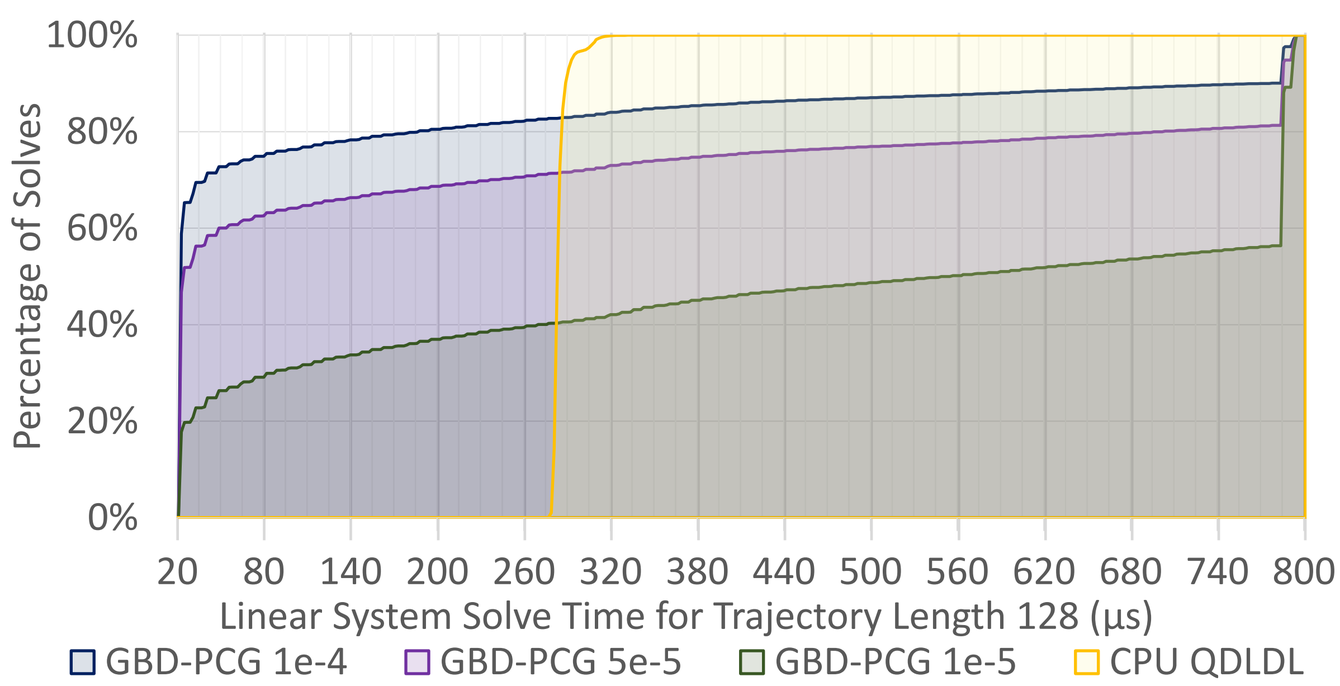
- GBD-PCG's advantage **scales with problem size, with up to a 3.6x average speedup** over QDLDL on the CPU.
- GBD-PCG, under multiple different exit tolerances, ϵ, exhibits a **bi-modal solve time distribution** which is usually much faster than the uni-modal distribution for QDLDL on the CPU. E.g., for $\epsilon = 1e^{-4}$:
  - **>65% of GBD-PCG solves are ≥10x faster** than the fastest QDLDL solve,
  - **<10% of GBD-PCG solves are ≥2x slower,** and the slowest is 2.5x slower, than the slowest QDLDL solve.
- Our **GPU-first approach** enables MPCGPU to scale **to 512 knot points at 1kHz** and execute 8 iterations for **128 knot points** at 500Hz, for a per-iteration rate of **4kHz**.

## GPU Implementation Insights:

- **Asynchronous floating point atomic operations** on the GPU **produce small numerical inconsistencies**, e.g., O(1e-7), which can negatively impact overall stability and convergence.
- **Warp-level tiling** enables fine-grain synchronization and concurrent heterogenous operations within kernel blocks.

| MPCGPU with GBD-PCG | | Knot Points | | | | |
|---|---|---|---|---|---|---|
| | | 32 | 64 | 128 | 256 | 512 |
| Control Rate | 250Hz | 22.2 | 19.7 | 15.4 | 5.2 | 4.4 |
| | 500Hz | 10.3 | 10.6 | 8.0 | 4.6 | 3.0 |
| | 1kHz | 4.9 | 5.2 | 3.7 | 2.4 | 1.7 |




This material is based upon work supported by the National Science Foundation (under Award 2246022). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the funding organizations.

*These authors contributed equally to this work.