Consider different sizes of square tracked

Slack constraints increase computation time – might not be worth it

From tuning

Trade-off between angle constraints and position constraints. When angle cost is high, crane will not reach the target state, when position cost is high, angles go wild

Increased angle constraint makes the pendulum follow a straighter line – probably because the controller is not trying too hard to keep it small hence fewer perturbations

Crane doing small movements to counter oscillations of the pendulum (seen in the graphs) – this might not be possible in the real hardware due to stickiness – would have to be tested and tuned

Non-constrained RHC just slightly worse – larger angle deviation in the corners (angle constraints help here). In reality, this might not be true, since crane might not be able to do fine movements to cancel angles around target state – hence constrained RHC would not be worth it.

Longer horizon length makes the controller stop sooner (i.e. before reaching target) – results in larger oscillation

Linear simulation much more sensitive to tight constraints – can run with only about 0.03 margin, while non-linear can handle down to 0.001 (it’s shit, but it runs)

After longer time (30 seconds) the non-lin simulation achieves steady state – the trace each square is identical