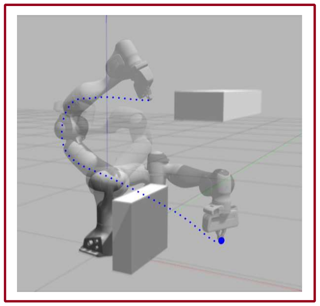
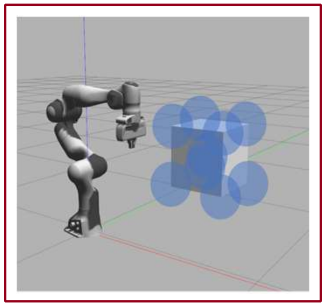
Model Predictive Control approaches for collision free manipulator arm control

Archit Hardikar , Joshua Ernest P, Rohit Bhikule

INTRODUCTION

- Numerous industrial applications involving robotic manipulators, it is imperative to develop efficient and safe motion control strategies for generating collision free trajectories.
- Model Predictive Control (MPC) is one of the most efficient point-to-point trajectory generation method used in situations that necessitate shortest execution time. It is a robust option for trajectory generation since its formulation includes a final-state constraint to be satisfied at the end of horizon while respecting states and inputs' limits.





BACKGROUND

- In general, applied methods for trajectory generation in robotics applications can be divided into two methods: Sampling-based and control based which have limitations such as producing jerky motions and having low-dimensional configuration space.
- Hence, there is a need for formulation of an optimization problem which considers all the required conditions and constraints. Therefore, the concept of MPC in a receding horizon formulation is suitable for solving the trajectory generation problem.
- Very less work has been done on finding the control policy in an obstacle cluttered environment solely by Linear MPC.
- Relevant work includes solving the problem using trajectory generation with MPC solely for control by the following two approaches - solving the optimization problem on a subset of the collision free state space and adding collision avoidance constraints to the optimization problem.
- Existing approaches of robot control MPC, do not guarantee a collision free trajectory in dynamic environments.

METHODS

$$x_c = [q_1 \quad \dot{q}_1 \quad \ddot{q}_1 \quad \cdots \quad q_7 \quad \dot{q}_7 \quad \ddot{q}_7] \in \mathbb{R}^{21}$$

$$u_c = \dddot{q} \in \mathbb{R}^7$$

$$\dot{x}_c(t) = A_c x_c(t) + B_c u_c(t)$$

$$V_k(U_k) = \sum_{j=k+1}^{k+H} x_j^T Q x_j + \sum_{j=k}^{k+H-1} u_j^T R u_j, \; Q \in \mathbb{R}^{21x21}, R \in \mathbb{R}^{7x7}$$

Hierarchical MPC-

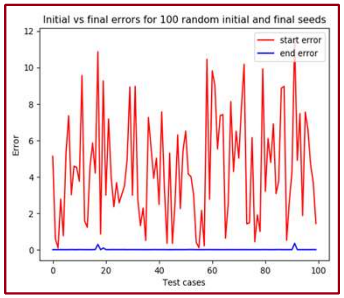
- In an obstacle cluttered environment, we used Potential field and RRT to generate waypoints. 21 variable Linear MPC was used for trajectory smoothing and rapidly providing efficient control policy. MPC was used to choose one from multiple RRT paths joining the start and target. The RRT path was optimized by finding the optimal cost.

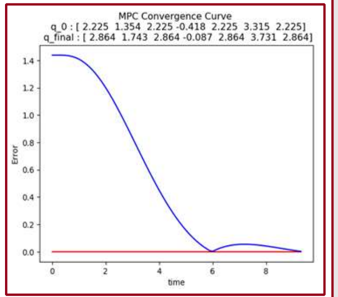
MPC for obstacle avoidance and control-

- Finding collision free trajectory is a non-convex problem. We transformed the 3D obstacle constraints to 7D joint space vector. Later, inverse kinematics was used to generate feasible robot configurations that reached the key features on the boundary of the obstacle. We randomly sampled 50 poses for each feature. Linear inequality constraints were imposed on robot state configuration to help avoid obstacles.
- We tried multiple approaches to solve this non-convex problem including defining convex sets outside obstacle. (See results)

RESULTS

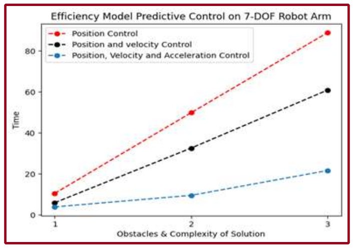
- We tested convergence of our MPC algorithm by iterating over 100 valid start and end joint configurations within the joint limits. It converged successfully for 197/200 configurations.

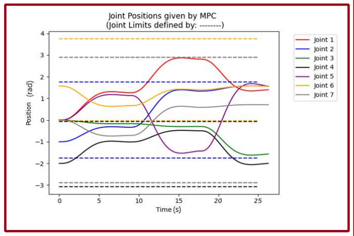


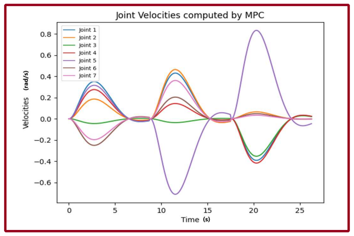


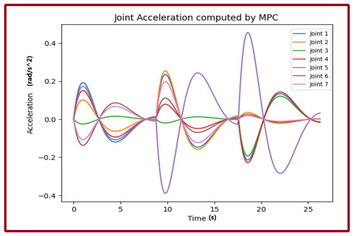
RESULTS Continued

- We evaluated the efficiency of the control policy determined by MPC+RRT pipeline in ROS by defining multiple complex obstacles in the reachable workspace.
- We also tried to impose constraints directly in the MPC formulation to help the algorithm avoid obstacles without the need of RRT but were unsuccessful owing to the problem's non convexity.
- Redefining the state space vectors and system continuous and discrete dynamics to include the joint positions in XYZ was not only difficult, but also increased the complexity of the MPC problem.
- We plan to reformulate our MPC to include the joint positions, which will significantly simplify the task of defining the obstacle in 7D space. We also plan to improve the robustness of our obstacle-aware MPC by better defining convex sets around the obstacle.
- Current obstacle-aware MPC fails to avoid obstacles on multiple occasions, because the obstacle is not well defined, and the constraints aren't sufficient.









REFERENCES

[1] Julian Ducaju et al., Joint Stiction Avoidance with Null-Space Motion in Real-Time Model Predictive Control for Redundant Collaborative Robots, 2021 30th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)