0.1 MIQP and Clustered MIQP

We wanted to compare the solutions provided by FOM with traditional MPC solving techniques. In order to do that we decided to implement the MIQP technique using the commercial solver Gurobi. This solver was already used in Hogan's work in a force-based formulation prior to the final velocity-based one (TODO: Cite) and it was able to run in an average of 3Hz. It should be mentioned that the execution time at every control iteration could range between few miliseconds to 10 seconds, because MIP algorithms depend on the geometry of the problem. This is not a problem for simulation, but in a real control environment the frequency at which you can update the controller cannot be higher that the frequency at which you can solve the worst case scenarios in the optimization problem. Otherwise the controller may try to update the control input before it is computed by the optimizer. There may be solutions to this effect in other formulations, but for the one used in Hogan's work an MIQP controller could not be run at more than 0.1Hz.

We were expecting a similar behavior in the velocity-based formulation, but it took several orders of magnitude more to solve. In order to simulate 500 control iterations (a trajectory of 5 seconds controlled at 100Hz) with a finite horizon of 7 steps it took around 6 hours of execution time. We were expecting to have an "ideal" MIQP result to compare FOM with and try to get similar results with FOM with less computational effort, but the results obtained with 7 steps of the MIQP were worse than the FOM and we couldn't afford to increase the horizon further.

We looked at techniques to speed up the MIQP execution (TODO: Add where did I find the information) and we found out that our current big-M formulation worked better (both in execution time and in quality of the obtained trajectory) than the indicator constraints formulations as smaller the M parameter was. As explained in **??** each M parameter used in a scalar equality $Ax \leq b + M(1-z)$ must be greater than $\max\limits_{x \in D}(Ax - b)$, for a the x variable domain D, so that the inequality is rendered redundant when $z = 0$. We implemented a version of MIQP that given a vectorial constraint as the ones used in Hogan's formulation $A\vec{x} \leq \vec{b} + M(1-z) \cdot (1, ..., 1)^T$ replaces the arbitrary constant M by a vector $\vec{M}$ where each component's value is $M_i = \max\limits_{x_i \in D}(A_i x_i - b_i)$. We were not able to improve the execution time, but we were able to obtain better results for the MIQP. TODO: add images showing the improvement.

In order to compare FOM to other methods using a longer time horizon, we implemented another MPC solving method based on MIQP that we called Clustered MIQP. Given a standard MPC formulation, this method works in the following way:

- Consider a clustering factor c, a finite horizon Nc and a set of h hybrid states $\{H_1, ..., H_h\}$.
- The finite horizon is then divided into N clusters of c steps: $(C_1, ..., C_N)$.
- The state space is extended with 3 binary variables per step (s_i, d_i, u_i) for a total of 3N variables.
- Each variable s_i d_i and u_i is associated with the hybrid state of the cluster C_i (sticking, sliding down and sliding up respectively).
- The constraint $s_i + d_i + u_i = 1$ is enforced
- The problem constraints are formulated with big M notation or indicator constraints and the problem is fed into an MIQP solver.

This way we can reduce the dimensionality of the optimization problem by keeping the same state during all the steps of the same cluster. The execution time of this new problem is similar to the excution time of a MIQP problem of N steps but the controller inputs can be different at each of the Nc steps.

We intended to use this method to approximate MIQP problems with long horizons (around 35 steps) by using few clusters (around 5) and we expected to obtain a better trajectory than the ones obtained with FOM. As will be discussed later, the results of this MIQP approximation were worse than the results obtained in FOM with a good choice of few modes.