Real-time model predictive control of a motion simulator based on cable robot technology

Mikhail Katliar

Systems Control and Optimization laboratory

SYSCOP group retreat
September 5-7, 2016

MPI CableRobot Motion Simulator

Video

Sensors and Actuators

- Actuators
  - 8 winches controlled by motor current
- Direct sensors
  - 8 cable tension sensors located in pulley axes
  - 8 motor position sensors (encoders)
  - 1 IMU on the platform

OCP formulation: dynamics of the motion system

- Rigid body dynamics with 8 cable forces and gravity force.
- Additional external force and torqe as a disturbance.
- Assume that the cable forces are directly controlled.
- Cables mass, cables elasticity, pulleys and winches dynamics, friction forces are not modelled.
- System state is described by the position of center of mass $\mathbf{r}$, orientation quaternion $\mathbf{q}$, velocity of center of mass $\mathbf{v}$ and rotational velocity ω:

$$\mathbf{x} = \begin{bmatrix} \mathbf{r} \\ \mathbf{q} \\ \mathbf{v} \\ \omega \end{bmatrix}$$

System dynamics: ODE

$$
\begin{aligned}
\dot{\mathbf{r}} &= \mathbf{v} \\
\dot{\mathbf{v}} &= \mathbf{g} + \frac{1}{m}\left(\sum_i \mathbf{F}_i + \mathbf{F}_{\text{ext}}\right) \\
\dot{\omega} &= I^{-1}\left(\sum_i \mathbf{b}_i \times \left(R(\mathbf{q})^\top \mathbf{F}_i\right) + \tau_{\text{ext}} - \omega \times (I\omega)\right) \\
\dot{\mathbf{q}} &= \frac{G(\mathbf{q})^\top}{2}\omega \\
\mathbf{F}_i &= \frac{\mathbf{l}_i}{\|\mathbf{l}_i\|} f_i \\
\mathbf{l}_i &= \mathbf{a}_i - \mathbf{r} - R(\mathbf{q})\mathbf{b}_i
\end{aligned}
\tag{1}
$$

$$
G(\mathbf{q}) = \begin{bmatrix} -q_1 & q_0 & q_3 & -q_2 \\ -q_2 & -q_3 & q_0 & q_1 \\ -q_3 & q_2 & -q_1 & q_0 \end{bmatrix}
$$

- $R(\mathbf{q})$ – rotation matrix from platform to world frame
- I – inertia tensor, m – mass
- $\mathbf{l}_i$ – vector connecting ends of i-th cable
- $\mathbf{b}_i, \mathbf{a}_i$ – coordinates of anchor point and outlet point of i-th cable
- f_i – tension force of i-th cable
- $\mathbf{F}_{\text{ext}}$ – external force
- τ_{ext} – external torque.

OCP formulation: path constraints

- Position limits

$$\mathbf{r}_{\text{min}} \leq \mathbf{r} \leq \mathbf{r}_{\text{max}}$$

- Linear velocity limit

$$\|\mathbf{v}\| \leq v_{\text{max}}$$

- Rotational velocity limit

$$\|\omega\| \leq \omega_{\text{max}}$$

- Cable force limits

$$0 < f_{\text{min}} \leq f_i \leq f_{\text{max}}$$

- Unit norm of quaternion (consistency constraint)

$$\|\mathbf{q}\| = 1$$

OCP formulation: terminal constraints

- For safety and stability reasons, the platform is required to stop at the end of the horizon:

$$\mathbf{v}(T) = 0$$
$$\omega(T) = 0$$

OCP formulation: terminal constraints

- For safety and stability reasons, the platform is required to stop at the end of the horizon:

$$\mathbf{v}(T) = 0$$
$$\omega(T) = 0$$

- Possible additional constraint: upright platform position at the end of the horizon:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} R\left(\mathbf{q}\left(T\right)\right)^{\top} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0$$

OCP formulation: objective function

- The objective function

$$\min_{\mathbf{x},\mathbf{u}} \quad \sum_{k=0}^{N-1} \left(\|\mathbf{u}_k - \hat{\mathbf{u}}_k\|^2_{W_\mathbf{u}} + \|\mathbf{x}_k - \hat{\mathbf{x}}_k\|^2_{W_\mathbf{x}} + \|\mathbf{y}(\mathbf{x}_k, \mathbf{u}_k) - \hat{\mathbf{y}}_k\|^2_{W_\mathbf{y}} \right) + \|\mathbf{x}_N - \hat{\mathbf{x}}_N\|^2_{W_{\mathbf{x}_N}}$$

s.t.

$$\mathbf{x}_{k+1} = F(\mathbf{x}_k, \mathbf{u}_k) \ \forall k = 0 \dots N-1$$

based on (1) and assuming $\mathbf{F}_{\text{ext}} = 0, \ \tau_{\text{ext}} = 0$

and the path and terminal constraints above

where $\mathbf{x}$ – system state, $\mathbf{u} = \left[f_1, f_2, \dots, f_{N_C}\right]^\top$ – system input, $\mathbf{y}(\cdot, \cdot)$ – output function, $\hat{\mathbf{u}}$, $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ – reference input, state and output, respectively.

OCP formulation: objective function

- The objective function

$$\min_{\mathbf{x},\mathbf{u}} \quad \sum_{k=0}^{N-1} \left(\|\mathbf{u}_k - \hat{\mathbf{u}}_k\|^2_{W_\mathbf{u}} + \|\mathbf{x}_k - \hat{\mathbf{x}}_k\|^2_{W_\mathbf{x}} + \|\mathbf{y}(\mathbf{x}_k, \mathbf{u}_k) - \hat{\mathbf{y}}_k\|^2_{W_\mathbf{y}} \right) + \|\mathbf{x}_N - \hat{\mathbf{x}}_N\|^2_{W_{\mathbf{x}_N}}$$

s.t.

$$\mathbf{x}_{k+1} = F(\mathbf{x}_k, \mathbf{u}_k) \; \forall k = 0 \ldots N-1$$

based on (1) and assuming $\mathbf{F}_\text{ext} = 0, \; \tau_\text{ext} = 0$

and the path and terminal constraints above

where $\mathbf{x}$ – system state, $\mathbf{u} = \left[f_1, f_2, \ldots, f_{N_C}\right]^\top$ – system input, $\mathbf{y}(\cdot, \cdot)$ – output function, $\hat{\mathbf{u}}$, $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ – reference input, state and output, respectively.

- The blue term is the one relevant for motion simulation.

OCP formulation: objective function

- The objective function

$$\min_{\mathbf{x},\mathbf{u}} \sum_{k=0}^{N-1} \left(\|\mathbf{u}_k - \hat{\mathbf{u}}_k\|^2_{W_\mathbf{u}} + \|\mathbf{x}_k - \hat{\mathbf{x}}_k\|^2_{W_\mathbf{x}} + \|\mathbf{y}(\mathbf{x}_k, \mathbf{u}_k) - \hat{\mathbf{y}}_k\|^2_{W_\mathbf{y}} \right) + \|\mathbf{x}_N - \hat{\mathbf{x}}_N\|^2_{W_{\mathbf{x}_N}}$$

s.t.

$$\mathbf{x}_{k+1} = F(\mathbf{x}_k, \mathbf{u}_k) \; \forall k = 0 \ldots N-1$$

based on (1) and assuming $\mathbf{F}_{\text{ext}} = 0, \; \tau_{\text{ext}} = 0$

and the path and terminal constraints above

where $\mathbf{x}$ – system state, $\mathbf{u} = \left[f_1, f_2, \ldots, f_{N_C} \right]^\top$ – system input, $\mathbf{y}(\cdot,\cdot)$ – output function, $\hat{\mathbf{u}}$, $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ – reference input, state and output, respectively.

- The blue term is the one relevant for motion simulation.
- Choosing weights $W_\mathbf{u}$, $W_\mathbf{x}$ and $W_\mathbf{y}$ allows to balance between input tracking, state tracking and output tracking.

OCP formulation: output function

- In motion simulation, we want to reproduce specific force $\mathbf{f}_H$, rotational velocity ω_H and rotational acceleration α_H in the reference frame attached to subject's head:

$$\mathbf{y}(\mathbf{u},\mathbf{x}) = \begin{bmatrix} \mathbf{f}_H \\ \omega_H \\ \alpha_H \end{bmatrix}$$

$$\mathbf{f}_H = R_P^H \left(R(\mathbf{q})^\top (\mathbf{g} - \dot{\mathbf{v}}) - \underbrace{\dot{\omega} \times \mathbf{r}_H}_{\text{Euler acceleration}} - \underbrace{\omega \times (\omega \times \mathbf{r}_H)}_{\text{centrifugal acceleration}} \right)$$

$$\omega_H = R_P^H \omega$$

$$\alpha_H = R_P^H \dot{\omega}$$

where $\mathbf{r}_H$ – position of the head in platform frame, R_P^H – rotation matrix from head frame to platform frame.

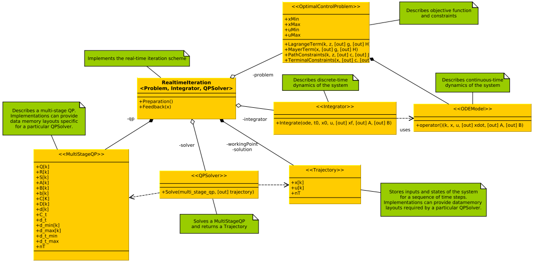
- Notice that the output directly depends on input ("direct feedthrough"), because of $\dot{\mathbf{v}}$ and $\dot{\omega}$ (look at (1)).

MPC controller properties

Table: MPC controller properties

HW sampling time	$\approx$ 1 ms
Control sampling time	50 ms
Prediction horizon (steps) N	40
Number of states	13
Number of controls	8
Integrator	Explicit RK4
Hessian approximation	Gauss-Newton
QP solver	qpOASES+condensing; HPMPC
Problem-specific code	Generated by CasADi
Implementation language	C++

Software implementation



- Based on *tmpc: Templates for Model Predictive Control* `http://gitlab.syscop.de/mikhail.katliar/tmpc`
- Unified interface to QP solvers, integrators etc. $\Rightarrow$ different controller implementations which use different components can be easily created.

MPC controller performance evaluation

Table: MPC controller performance evaluation

CPU type	AMD A8-4500M @1.9 GHz
Preparation phase time	1.9 ms
Feedback phase time with qpOASES+condensing (avg.)	820 ms
Feedback phase time with HPMPC (avg.)	8.5 ms

- The controller can run at almost 100 Hz on my laptop
- 20 Hz is the required minimum for motion simulation

MPC controller performance evaluation

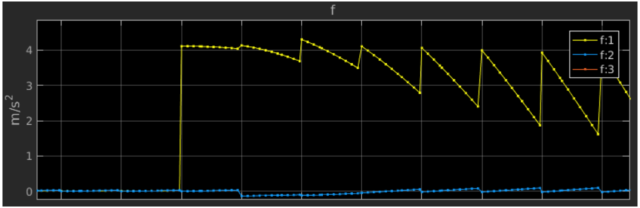
Table: MPC controller performance evaluation

CPU type	AMD A8-4500M @1.9 GHz
Preparation phase time	1.9 ms
Feedback phase time with qpOASES+condensing (avg.)	820 ms
Feedback phase time with HPMPC (avg.)	8.5 ms

- The controller can run at almost 100 Hz on my laptop
- 20 Hz is the required minimum for motion simulation
- HPMPC has made it possible (thanks Gianluca)!

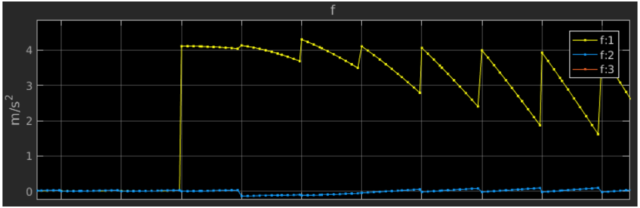
Issues: discontinuous output

- Due to direct feedthrough, for a piecewise-continuous input the output is also only piecewise-continuous:



Issues: discontinuous output

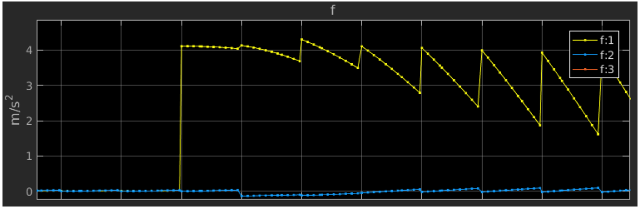
- Due to direct feedthrough, for a piecewise-continuous input the output is also only piecewise-continuous:



- The output changes significantly by the end of a sampling interval, but only the output value at the beginning enters the objective function $\Rightarrow$ poor output tracking.

Issues: discontinuous output

- Due to direct feedthrough, for a piecewise-continuous input the output is also only piecewise-continuous:



- The output changes significantly by the end of a sampling interval, but only the output value at the beginning enters the objective function $\Rightarrow$ poor output tracking.
- Possible remedies:
  1. Make input continuous by controlling cable forces change rate rather than cable forces themselves (IMPLEMENTED).
  2. Accurately integrate output error instead of evaluating it at one point per interval (WORTH IMPLEMENTING?).

Demo

Now time for a demo!

Thank you!

Thank you very much for your attention!